

# Budget Management

Catalog Number 26-1603


**Radio Shack®**  
**TRS-80**  
**MICRO**  
**COMPUTER**  
**SYSTEM**

**SOFTWARE**

**Budget Management allows you to keep track of up to:**

- **60 Different Budgeted Accounts**
- **25 Cash Expenses Per Month**
- **20 Savings Deposits or Withdrawals Per Month**
- **50 Checks Per Month**
- **20 Deposits Per Month**

**You can also keep track of cancelled checks. With this information instantly at your disposal, you can easily set up and then stick to a realistic budget.**

CUSTOM MANUFACTURED IN USA BY RADIO SHACK  A DIVISION OF TANDY CORPORATION

FIRST EDITION -- 1979

All rights reserved. Reproduction or use, without express permission, of editorial or pictorial content, in any manner, is prohibited. No patent liability is assumed with respect to the use of the information contained herein. While every precaution has been taken in the preparation of this book, the publisher assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained herein.

© Copyright 1979 Radio Shack,  
A Division of Tandy Corporation,  
Fort Worth, Texas 76102, U.S.A.

Printed in the United States of America

# **BUDGET MANAGEMENT**

For the Radio Shack TRS-80  
LEVEL II BASIC  
or  
DISK BASIC

**Radio Shack®**  
A DIVISION OF TANDY CORPORATION

One Tandy Center  
Fort Worth, Texas 76102





# Contents

SETTING UP .....	3
Required Equipment, Uses of Program, Loading Under LEVEL II, Loading Under DISK BASIC.	
INITIALIZATION .....	9
Setting Up Accounts, Creating Accounts, Saving Accounts, Inserting Accounts, Renumbering Accounts, Finishing Up.	
TRANSACTION .....	21
When and How To Use, Deposit Checkbook, Outstanding Checks, Cash Expenses, Income Accounts, Savings Accounts, Transfer Funds, Cancel Checks, Finishing Up.	
EDIT .....	31
Correcting Mistakes, Edit Accounts, Edit Outstanding Checks, Edit Cancelled Checks, Edit Cash Expenses, Edit Deposits, Edit Savings, Finishing Up.	
STARTING A NEW MONTH .....	37
How to Start a New Month of Transactions.	
REPORTS .....	39
To View Results, Print Monthly Reports, Sample Reports, Print Cumulative Reports, Tips on Reading Reports.	
APPENDIX .....	51
Worksheets, Generating Your Own Account List, Using a Quick Printer, Error Messages, Variable Usage, Memory Usage, Program Listings.	



### What Do I Get? . . .

The four programs in this package are on two cassette tapes. If you have a TRS-80 Disk System, you can easily convert them to a diskette. Instructions are included in this manual.

### . . . And What Must I Already Have?

The basic equipment you'll need depends on the kind of system you have.

#### LEVEL II BASIC

16K RAM (minimum)  
One Cassette Recorder (minimum)  
Printer (optional)

#### DISK BASIC

32K RAM (minimum)  
One Disk Drive (minimum)  
Printer (optional)

The Budget Management Package will *automatically* adjust itself to use tapes on a 16K LEVEL II BASIC computer, or disk files on a 32K DISK BASIC computer.

The optional printer is needed only if you want to print reports. You can use a line printer, a Radio Shack Quick Printer, or a TRS-80 Screen Printer to print reports. Otherwise, the reports will be displayed on your Video Screen.

Of course, you'll need blank cassettes or diskettes to store your budget information.

But you don't need a lot of knowledge. That's what this manual is for.

### But What Good Is It?

If you wonder where all your money is going . . . and who doesn't . . . or if you're trying (unsuccessfully) to stick to a budget that's not realistic . . . then you'll really appreciate the Radio Shack Budget Management Package.

This package has four programs designed especially to help you set up and monitor your own personal budget. Of course, you'll still have to supply the will power to stick to that budget. But we hope these programs will help you to supply that will power.

And best of all! You don't have to be a computer expert to use Budget Management. You don't even need to know BASIC (the language your Computer "speaks"). (If you're already familiar with LEVEL II or DISK BASIC, you'll be interested in the program listings in the back of this manual.)

Budget Management isn't an Accounting Package, though, it's for budgeting only. Accounting is more business-like and legal. For example, under Budget Management, you set up house payments as an expense account. In Accounting, house payments are partly an asset, and partly an expense.

So to put it in a nutshell, Budget Management will help you make decisions on where to spend your money. It keeps lots of facts organized for you so it's easy to see where you stand. If you need a business Accounting Package, you should get Radio Shack's General Ledger I.

### Is There Anything Else?

One more thing! A lot can happen to destroy cassettes or diskettes and the information stored on them.

Since these programs store all your accounts on tape or diskette, you ought to make it a routine to keep extra copies of important files — not only here but in all your computer work. We'll make suggestions when files are important enough to create backup copies.

## **OK, I'm Convinced . . . Now When Can I Use It?**

You can begin right away.

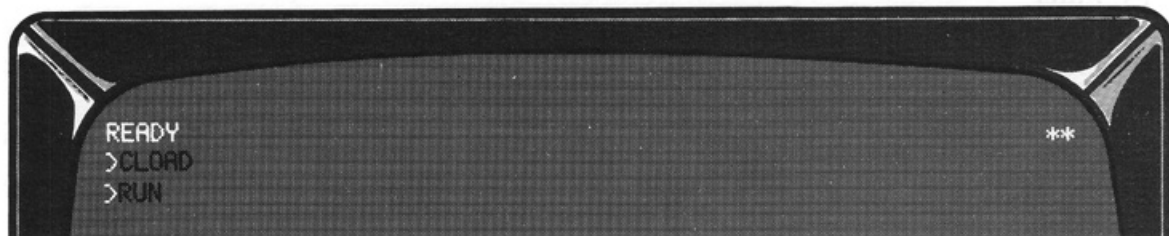
INITialization is the first program that you will use. It creates your account structure, inserts new accounts, and renumbers your account numbers, but you won't use it very often after this. Later you'll need it only if you want to drastically reorganize your accounts.

## **What Do I Do First?**

The first thing you do is load the INIT program into memory. Instructions for loading programs are the same no matter which of the four programs you will use. Responses that you must type are highlighted in black or gray.

## **For LEVEL II BASIC Only**

1. Make sure your recorder is properly connected and BASIC is running.
2. Place the INIT cassette in your recorder and push the Play key.
3. Type **CLOAD** **ENTER**. The computer will read in the program automatically. You can be sure this is happening when two asterisks appear (one of them blinking) in the upper right of your Video Display.
4. When the program has been completely read in, the cassette will stop and the **READY** prompt will return. Type **RUN** **ENTER** and the program will begin.



## SETTING UP

---

### For DISK BASIC Only

If you have DISK BASIC, there are a few more steps to go through, but they are all easy. But following all these instructions, you can save the programs on diskette and run them more easily later.

1. Make a backup copy of TRSDOS and DISK BASIC on a blank diskette using the TRSDOS utility BACKUP. (See the TRSDOS manual for detailed instructions.)
2. Using this diskette, load TRSDOS. Type **AUTO VERIFY** **ENTER**. Load DISK BASIC by typing **BASIC** **ENTER**. Press **ENTER** to answer the Files and Memory Size questions.\*
3. Type **CMD "T"** **ENTER**.
4. Place the INIT cassette in your recorder. (Make sure your recorder is properly connected, and the Play key is pushed.)
5. Type **CLOAD** **ENTER**. The INIT program will automatically load into memory (and two asterisks will appear in the upper right corner of your Video Display.)
6. Type **CMD "R"** **ENTER**.
7. Type  
**5000 IF E=99 PRINT "\*\*END OF RUN": RUN "MENU"**
8. To make sure you typed this correctly, type  
**LIST 5000** **ENTER**.
9. If it's OK, type **SAVE "INIT"** **ENTER**.
10. Use the TRANS tape and repeat steps 3-6.
11. Enter  
**5500 IF E=99 PRINT "\*\*END OF RUN": RUN "MENU"**
12. Type **LIST 5500** **ENTER** to check this.
13. If it's OK, type **SAVE "TRANS"** **ENTER**.
14. Use the EDIT tape and repeat steps 3-6.
15. Enter  
**5500 IF E=99 PRINT "\*\*END OF RUN": RUN "MENU"**
16. Type **LIST 5500** **ENTER** to check this.
17. If it's OK, type **SAVE "EDIT"** **ENTER**.
18. Use the REPORT tape and repeat steps 3-6.
19. Enter  
**5000 IF E=99 PRINT "\*\*END OF RUN": RUN "MENU"**
20. Type **LIST 5000** **ENTER** to check this.
21. If it's OK, type **SAVE "REPORT"** **ENTER**.

\*48K SYSTEMS ONLY – SET MEMORY SIZE TO 65529

---

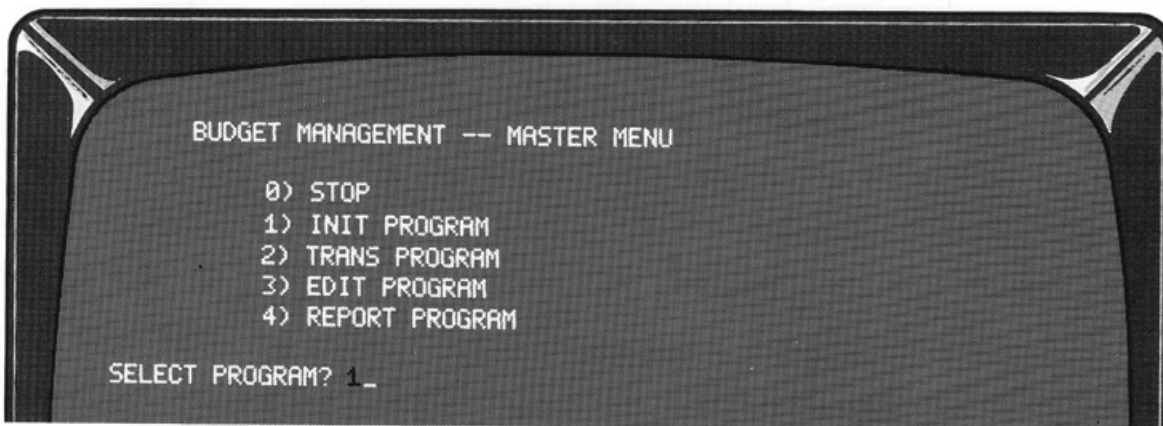
22. Enter the following lines exactly as they appear below:

```
NEW
100 'BUDGET MANAGEMENT — MASTER MENU 1.0
110 '(C.) 1979 TANDY CORP.
120 CLS: PRINT TAB(15) "BUDGET MANAGEMENT — MASTER MENU":PRINT
130 PRINT TAB(20)"0.) STOP":PRINTTAB(20) "1) INIT PROGRAM
140 PRINT TAB(20) "2.) TRANS PROGRAM":PRINT TAB(20) "3.) EDIT PROGRAM
150 PRINT TAB(20) "4) REPORT PROGRAM":PRINT
160 PRINT TAB(10) "SELECT PROGRAM":INPUT J
170 ON J+1 GOTO 200, 300, 400, 500, 600:GOTO 120
200 PRINT "'* END OF SESSION *':END
300 RUN "INIT"
400 RUN "TRANS"
500 RUN "EDIT"
600 RUN "REPORT"
```

23. Type **LIST** **ENTER**, and make sure that all lines are exactly as they appear above.

24. If they're OK, type **SAVE "MENU"** **ENTER**

To run the Budget Management Package, insert this Program diskette. Type **RUN "MENU"** **ENTER** and the following option list will be displayed.



**ENTER** the number of the program you want and it will automatically load and run. Whenever you finish with one of the four programs, you will be returned to this option list to choose another one. If you want to stop, select option 0 to go back to DISK BASIC.

For now, enter option 1 to run the INIT program.





### How Do I Create Accounts?

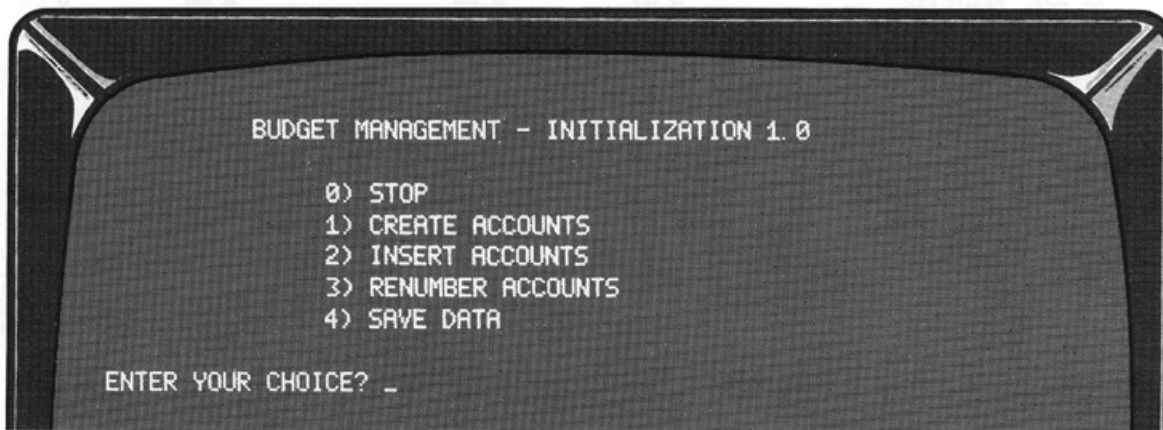
As soon as you run the INIT program, the following is displayed.

```
BUDGET MANAGEMENT - INITIALIZATION 1.0
```

```
ENTER TODAY'S DATE (MM/DD/YY)? 09/05/78_
```

Before you can do anything else, you must type a valid date. MM=the month. It must be two digits between 1 and 12. DD=the day, also two digits between 1 and 31. YY=the year. Use the last two digits of the current year. The month, day, and year are each separated from one another by a slash, /.

When you type a valid date, the following will be displayed.



Now, you can create your accounts. The next two pages show a listing of the default accounts already programmed into INIT. You should take some time to look over these and, using them as a guide, decide which accounts you need and what accounts you want to add to these. You can use the worksheets in Appendix A to help with this.

An account is simply one of the possible categories where you can get or spend money. All of the accounts together should reflect all of the areas where you can spend or receive money.

For example, your Checking Account is an account. So is your Income, your Savings Account, your Mortgage or Rent, Utilities, etc.

# INITIALIZATION

## BUDGET MANAGEMENT REPORT

PAGE 1

\*\* ACCOUNTS REPORT \*\*

DATE: 10/ 1/78

ACCOUNT =====	NAME =====	CURRENT =====	BUDGET =====	PERCENT =====	REMAINING =====
100	CHECKBOOK BALANCE .....	0.00	0.00	0.00	0.00
200	SAVINGS ACCOUNT .....	0.00	0.00	0.00	0.00
300	PETTY CASH .....	0.00	0.00	0.00	0.00
400	TOTAL INCOME .....	0.00	0.00	0.00	0.00
410	REGULAR INCOME .....	0.00	0.00	0.00	0.00
420	WAGE/SALARY .....	0.00	0.00	0.00	0.00
430	BUSINESS INCOME .....	0.00	0.00	0.00	0.00
440	SOC. SEC. BENIFITS .....	0.00	0.00	0.00	0.00
450	PENSION .....	0.00	0.00	0.00	0.00
460	VARIABLE INCOME .....	0.00	0.00	0.00	0.00
470	BONUS .....	0.00	0.00	0.00	0.00
480	INTEREST .....	0.00	0.00	0.00	0.00
490	SAVINGS .....	0.00	0.00	0.00	0.00
500	BONDS .....	0.00	0.00	0.00	0.00
510	DIVIDENDS .....	0.00	0.00	0.00	0.00
520	OTHER INCOME .....	0.00	0.00	0.00	0.00
600	SET-ASIDE INCOME .....	0.00	0.00	0.00	0.00
610	FUTURE GOALS .....	0.00	0.00	0.00	0.00
620	EMERGENCIES .....	0.00	0.00	0.00	0.00
630	EDUCATION .....	0.00	0.00	0.00	0.00
640	RETIREMENT .....	0.00	0.00	0.00	0.00
650	VACATION .....	0.00	0.00	0.00	0.00
700	TOTAL EXPENSES .....	0.00	0.00	0.00	0.00
710	REGULAR EXPENSES .....	0.00	0.00	0.00	0.00
720	RENT/MORTGAGE .....	0.00	0.00	0.00	0.00
730	HOUSE MAINTENANCE .....	0.00	0.00	0.00	0.00
740	UTILITIES .....	0.00	0.00	0.00	0.00
750	ELECTRICITY .....	0.00	0.00	0.00	0.00
760	NATURAL GAS .....	0.00	0.00	0.00	0.00
770	WATER .....	0.00	0.00	0.00	0.00
780	TELEPHONE .....	0.00	0.00	0.00	0.00
790	INSTALLMENT PAYMENTS .....	0.00	0.00	0.00	0.00
800	INSURANCE .....	0.00	0.00	0.00	0.00

# INITIALIZATION

BUDGET MANAGEMENT REPORT  
 \*\* ACCOUNTS REPORT \*\*  
 DATE: 10/ 1/78

PAGE 2

ACCOUNT	NAME	CURRENT	BUDGET	PERCENT	REMAINING
=====	=====	=====	=====	=====	=====
810	FOOD & BEVERAGE .....	0.00	0.00	0.00	0.00
820	FURNISHINGS .....	0.00	0.00	0.00	0.00
830	CLOTHING .....	0.00	0.00	0.00	0.00
840	TRANSPORTATION .....	0.00	0.00	0.00	0.00
850	BUS .....	0.00	0.00	0.00	0.00
860	CAR .....	0.00	0.00	0.00	0.00
870	GAS .....	0.00	0.00	0.00	0.00
880	CAR MAINTENANCE .....	0.00	0.00	0.00	0.00
890	MEDICAL .....	0.00	0.00	0.00	0.00
900	RECREATION .....	0.00	0.00	0.00	0.00
910	MISCELLANEOUS .....	0.00	0.00	0.00	0.00
920	SEASONAL EXPENSES .....	0.00	0.00	0.00	0.00
930	EDUCATION .....	0.00	0.00	0.00	0.00
940	CONTRIBUTIONS .....	0.00	0.00	0.00	0.00
950	TAXES .....	0.00	0.00	0.00	0.00
960	CHRISTMAS .....	0.00	0.00	0.00	0.00
970	BIRTHDAYS .....	0.00	0.00	0.00	0.00
980	TRAVEL .....	0.00	0.00	0.00	0.00

Using the worksheets in Appendix A, write down the accounts you need. Use these default accounts as a guide.

Indent your accounts to show the main account/sub-account hierarchy you want.

## INITIALIZATION

---

ACCOUNT NUMBER	LEVEL	ACCOUNT NAME	CURRENT	BUDGET
100	0	CHECKBOOK BALANCE	500	0
200	0	SAVINGS ACCOUNT	200	0
300	0	PETTY CASH	0	0
400	0	TOTAL INCOME		
410	1	REGULAR INCOME		
420	2	WAGE/SALARY	0	1200
600	0	SET-ASIDE INCOME		
610	1	FUTURE GOALS	0	50
620	1	EMERGENCIES	0	20
700	1	TOTAL EXPENSES		
710	1	REGULAR EXPENSES		
720	2	RENT/MORTGAGE	0	140
730	2	CAR PAYMENTS	0	105
740	2	UTILITIES	0	60
780	2	TELEPHONE	0	25
810	2	FOOD & BEVERAGE	0	150
830	2	CLOTHING	0	40

The above shows a worksheet that has been filled out.

Use the LEVEL column to show the indentation level of that account in the hierarchy.

## INITIALIZATION

---

There are two rules to keep in mind while filling this out. First, notice that in the default list some account names are indented under others. This indentation reflects a hierarchy between the accounts. A main or principal account can be divided into several sub-accounts.

```
84Ø TRANSPORTATION
85Ø   BUS
86Ø   CAR
87Ø     GAS
88Ø     MAINTENANCE
```

Above, **TRANSPORTATION** is divided into sub-accounts **BUS** and **CAR**. **CAR** is further divided into **GAS** and **MAINTENANCE**. In all cases, the main account total will equal the sum of the sub-accounts. So, enter totals only for the most indented level of sub-accounts under a given main account. In the example, you don't need to put a total for **84Ø** or **86Ø**, since both of these have sub-accounts.

**CAR = GAS + MAINTENANCE**  
**TRANSPORTATION = BUS + CAR**

The **Level** can be from 0 to 5 and is a hierarchy level for that account. Zero is the Level for main accounts only. A sub-account can be indented only one Level more than the previous account. For example, **REGULAR INCOME** cannot be Level two or three. It must be one.

The **CURRENT** value is the balance of an account. It is the amount of money spent or earned in that category so far. If it is the beginning of the month, most **CURRENT** values will be zero.

The **BUDGET** value is the amount you expect to spend or earn in that category during an average month.

The **ACCOUNT NAME** can be no more than 18 characters.  
**CAUTION: Do not use commas or colons in the ACCOUNT NAME.**

The **ACCOUNT NUMBER** follows from the default list.

**NOTE:** You only need to enter amounts for sub-accounts which have no sub-accounts of their own. In the example above, you don't need to enter an amount for **TOTAL EXPENSES** or **REGULAR EXPENSES** because both of these accounts have sub-accounts. Their amounts will be figured from the sum of their sub-accounts.

## INITIALIZATION

---

Now, the second rule you should keep in mind is that, whether you use the default list or not, the following five accounts must be present for the program to work properly.

**CHECKBOOK BALANCE**  
**SAVINGS ACCOUNT**  
**PETTY CASH**  
**TOTAL INCOME**  
**TOTAL EXPENSES**

The last two, **TOTAL INCOME** and **TOTAL EXPENSES** must have at least one sub-account to be of any use. If you use the default list (or a part of it) a sixth account **SET ASIDE INCOME** will also be used. It must also have sub-accounts.

**CHECKBOOK BALANCE** is, of course, your checking account. Its balance is the dollar value of the checks you can write.

The **SAVINGS ACCOUNT** is for your savings. Savings transactions affect the balance of this account.

The **PETTY CASH** account represents your cash on hand or pocket money. When you spend your pocket money, the balance of the **PETTY CASH** account is reduced.

Once you have decided on the accounts you need, there are two possible ways to feed your accounts to the Computer. You can use the default list if you have only a few accounts to add to it. Or create your own account structure and modify the program to accept it instead of the default list. (See Appendix B if you want to change the program.)

### Using the Default List

If you only have a few accounts to add, use the default list. You can insert your additions later. You can leave out as many of the accounts on the default list as you wish by following the instructions below.

Type **1** **ENTER** to choose the **CREATE ACCOUNTS** option. The default list will be displayed, and you should press **Y** if you want to use that account or **N** if not. (You won't be asked about the six main, level 0, accounts, they are not optional.)

If you answer **N** to an account, all of its sub-accounts will be skipped. So, if you want to use a sub-account, be sure to answer **Y** to its principal account.

## INITIALIZATION

If you press a key other than **Y** or **N** an error message will appear, and you will be allowed to try again. If you press **ENTER** without typing any other key, the program will assume the last answer you typed is also true for this account.

```
ACCOUNT INITIALIZATION
WANTED(Y/N)

100 CHECKBOOK BALANCE
200 SAVINGS ACCOUNT
300 PETTY CASH
400 TOTAL INCOME
410 REGULAR INCOME      ? Y
420 WAGE/SALARY         ? Y
430 BUSINESS INCOME     ? N
440 SOC. SEC. BENIFITS   ?
450 PENSION             ?
460 VARIABLE INCOME     ? N
600 SET-ASIDE INCOME
610 FUTURE GOALS        ? Y
620 EMERGENCIES         ? Y
630 EDUCATION           ? N
```

Notice that you must use accounts 100, 200, 300, 400, 600, and 700. By answering **N** to **VARIABLE INCOME**, all of its sub-accounts are skipped. Pressing only **ENTER** to

440 and 450 caused an answer of **N** to be assumed. Typing **R ENTER** to an account creates an error message and gives you the chance to try again.

```
ACCOUNT INITIALIZATION
CURRENT      BUDGET

100 CHECKBOOK BALANCE    ? 124.78  ?
200 SAVINGS ACCOUNT       ? 256.23  ?
300 PETTY CASH            ? 75.5    ?
400 TOTAL INCOME
410 REGULAR INCOME
420 WAGE/SALARY           ?          ? 1212
600 SET-ASIDE INCOME
610 FUTURE GOALS          ?          ? 20
620 EMERGENCIES           ? 0       ? 25
700 TOTAL EXPENSES
710 REGULAR EXPENSES
720 RENT/MORTGAGE         ?          ? 200
740 UTILITIES             ?          ? 60
```



## INITIALIZATION

---

After the account structure is set up, the INIT program starts over with the first account and allows you to enter the **CURRENT** and **BUDGET** amounts for the accounts you chose previously.

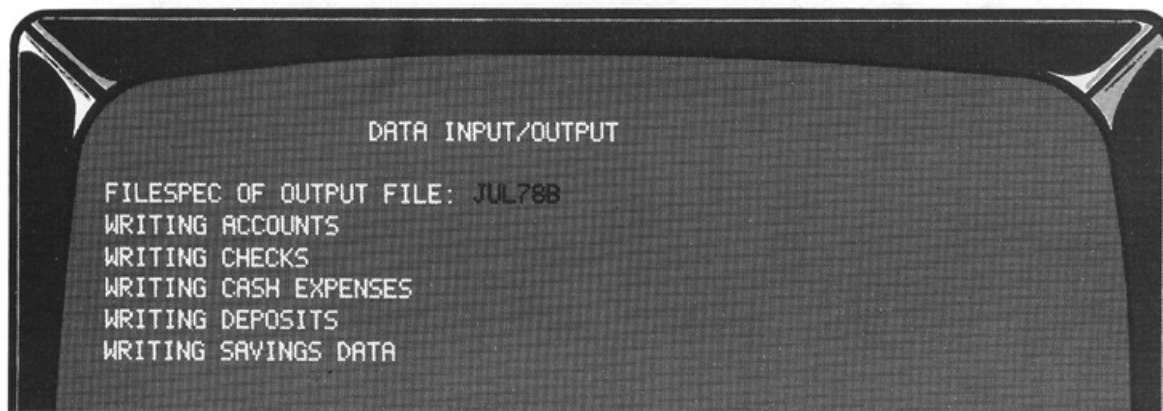
Pressing only **ENTER** causes the Computer to assume a value of zero. You will be returned to the option list automatically when finished.

If you have no accounts to add and your account structure is exactly the way you want it to be, then save the data by following the instructions below. Otherwise, you should insert the accounts you want before saving them. See page 17 for instructions on this.

### Saving Your Accounts

Once your accounts are completed, you must save them using option 4. All of the programs (except REPORT) must be saved since your accounts are stored in the Computer's temporary memory. If you turn your machine off, they will be lost. When you record them on cassette or diskette, you can play them back later.

If you have DISK BASIC, you will be asked for a TRSDOS filespec. Insert a TRSDOS diskette (or FORMAT a blank diskette for data if you have more than one drive). Type in a valid file name for your accounts and press **ENTER**. Do not use a period as part of the filespec unless you want to create a protected file. See below.



If you have only one drive replace the Program diskette with the System diskette that you will save data on. When the data is saved, put the Program diskette back in. If you

have 2 drives leave the Program diskette in Drive 0 and use a Data diskette in Drive 1. Specify a Drive Spec as part of the filespec.



Using LEVEL II, the following will be displayed.

LOAD DATA TAPE(RECORD), THEN PRESS ENTER?\_

Insert a blank cassette tape to save your data on and ready your recorder. Make sure the Record and Play keys are depressed. Press **ENTER**.

When the accounts have been saved, the option list will reappear. You should save a second, backup copy.

### Adding Accounts to the Default List

Below is an example of inserting the account 725 CAR PAYMENTS at an indentation level of 2. Type **→** to indent each level. CAR PAYMENTS was assigned account number 725 on the worksheet which puts it between 720 and 740.

```
ACCOUNT INSERTION

ENTER NEW ACCOUNT NUMBER? 830
* BAD ACCOUNT NUMBER *

ENTER NEW ACCOUNT NUMBER? 725

720    RENT/MORTGAGE          CURRENT    BUDGET
725    CAR PAYMENTS          ? 0        ? 105
740    UTILITIES

FIXING REFERENCE POINTERS...

ENTER NEW ACCOUNT NUMBER? END_
```

The account above and below the insertion point appears. Then, the new **ACCOUNT NUMBER** is printed between them. You can type the new required information, **ACCOUNT NAME** and **BALANCES**.

## INITIALIZATION

---

If the **ACCOUNT NUMBER** already exists or is invalid, a **\* BAD ACCOUNT NUMBER \*** message is displayed. If you already have the maximum of 60 accounts, a **\* NO ROOM TO INSERT \*** message occurs.

The cursor will sit at an indentation level of 1 and wait for you to enter your **ACCOUNT NAME**. Type a right arrow **→** to indent to the next level. If you make a mistake indenting, type a shift back-arrow **SHIFT ←** to start over again at level 1. To add a zero-level account, you must create your own account list. See Appendix B.

Type your **ACCOUNT NAME** after indenting as desired. Do not use commas or colons in the **ACCOUNT NAME**! The **ACCOUNT NAME** can be no greater than 18 characters. Press **ENTER**.

If you have indented more than 5 levels, or have indented more than one level inward from the account above your insertion point, a **\* BAD LEVEL \*** message is typed.

Press **ENTER** without typing anything to cancel the insertion in case the wrong **ACCOUNT NUMBER** was typed.

Type **END ENTER** instead of an **ACCOUNT NUMBER** to return to the option list.

After you enter the **ACCOUNT NAME**, you will be asked the **CURRENT** and **BUDGET** values. Press **ENTER** for a zero value.

NOTE: The **INSERT** option cannot change the main account/sub-account structure already in existence. In the example below, the insertion of 3250 CAR2 will not change the existing structure.

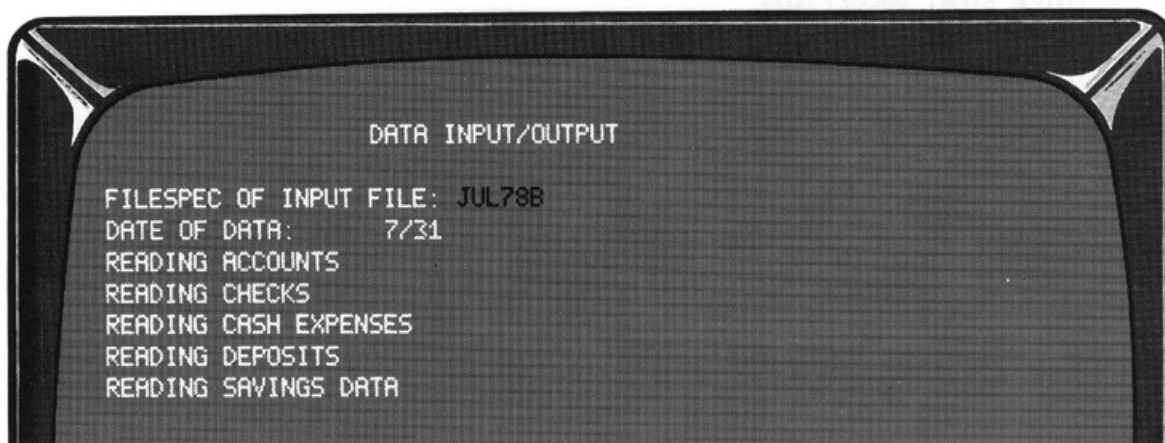
```
3000 TRANSPORTATION
3100  BUS
3200  CAR
3300    GAS
3400    MAINTENANCE
```

The structure will still have 3200 CAR as the main account to sub-accounts 3300 and 3400, even though they are separated by the 3250 account as shown below.

```
3000 TRANSPORTATION
3100  BUS
3200  CAR
3250  CAR2
3300    GAS
3400    MAINTENANCE
```

You can insert accounts later if you wish. However, this will affect the way you run cumulative reports. See notes on running these cumulative reports page 47.

To insert accounts later, run the INIT program and request option 2. You will be asked to load your most recent accounts from cassette or diskette. If you have DISK BASIC, the following will be displayed.



Under LEVEL II, you will be prompted with

```
LOAD DATA TAPE(PLAY), THEN PRESS ENTER?_
```

As soon as you have inserted the proper diskette or cassette and typed the requested information, press **ENTER** and the data will load into memory. Once there, you can insert new accounts as illustrated above.

NOTE: Using option 4, save your new account data once you have added what you want.

### Should I Renumber My Accounts?

Renumbering accounts is a very severe action and should only be done when room is needed to insert new accounts. Also, you can renumber when you first create accounts and before you save them to put them in neat numerical sequence.

If you renumber accounts in the middle of the year, you cannot create cumulative reports (account numbers must be the same to do this).

## INITIALIZATION

---

So, don't renumber your accounts if they are acceptable as they are.

If you do decide to renumber accounts, use option 3. You will be asked to load your accounts from cassette or diskette if they are not already in memory. Then, you will be asked to enter a **START VALUE** and an **INCREMENT**.

### ACCOUNT RENUMBERING

```
ENTER START VALUE? 100
ENTER INCREMENT? 100
RENUMBERING...
```

You should keep a record of the new account numbers on your worksheet next to the old numbers. From this time on, refer to the new numbers.

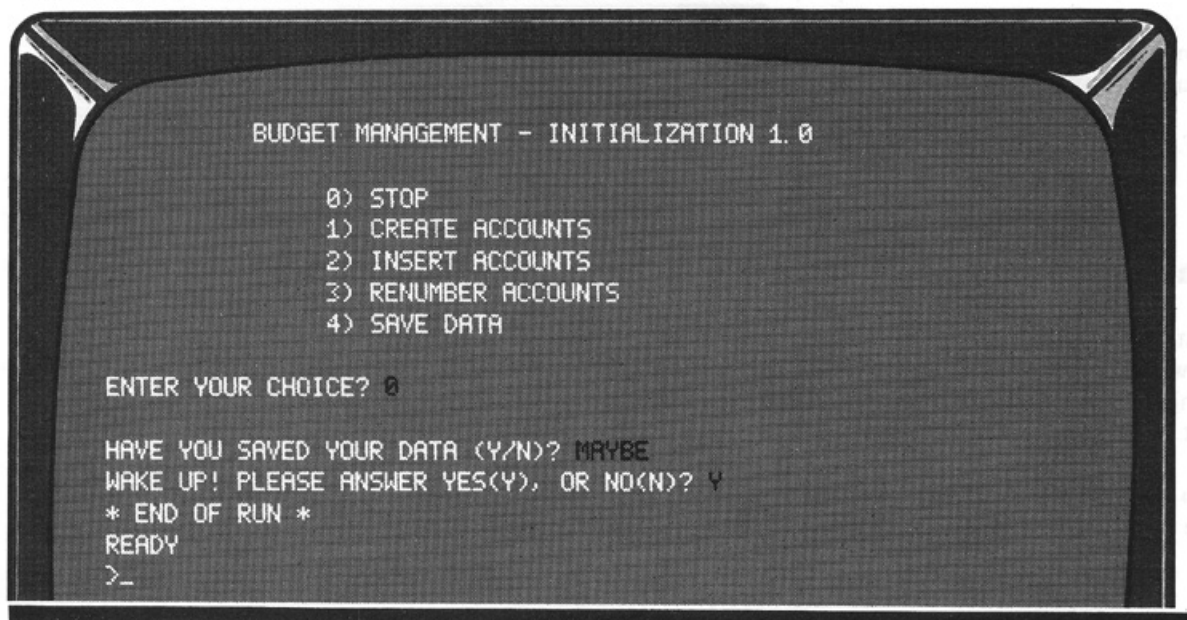
NOTE: Use option 4 to save your accounts with their new numbers.

### Is That All?

Once you have created your account structure as you want it — including insertion and renumbering — and you have saved this data, you can select the 0 option to stop this program. This option will ask you if you have saved your data as a reminder. If not, press **N**, and you will be able to save it now. If you have already saved it, press **Y**, and the **\*END OF RUN\*** will be displayed.

Under DISK BASIC, you will be returned to the MENU program so you can run another program if you want.

Under LEVEL II, you will be returned to BASIC.



### When Should I Use TRANS?

The TRANS program is the heart of the Budget Management Package. It allows you to enter and cancel checks, enter deposits and enter savings withdrawals or deposits. Also, it will clear your old transactions so you can start a new month of data. This last feature will be described later.

You should set up a timetable for using this program regularly. A few suggested schedules follow.

You can use the TRANS program as soon as you create your accounts. Then, you can run it once each week and again on the last day of the month. On this schedule, you will save one copy of your accounts after each run of TRANS. You can label or name these WEEK1, WEEK2, WEEK3, etc. They will be temporary files.

On the last day of the month, your accounts will reflect the entire month, and you should label them JAN79, FEB79, etc., depending on the month. You should save an extra copy of this file for backup purposes since you will be keeping it for some time.

Then, during the next month, you can re-use your temporary files from WEEK1, etc. as the temporary files of this month. You will have at least six sides of tapes in use after the first month. At the end of each succeeding month, you will have two more tapes. In DISK BASIC, use a diskette for these files and name them as illustrated above.

NOTE: If you record more than one data file on each side of a cassette, be sure to keep track of where each file begins using the counter on your cassette recorder.

An alternative schedule is to use the TRANS program only two times each month. You can label your accounts JAN79M for the mid-month temporary file and JAN79E for the end of month file that you save.

You could also run the program every other week or run it to coincide with your payday. But, whatever you decide for your schedule, you should include running the TRANS program at the end of each month and saving this copy permanently. And don't forget to keep backups.

## TRANSACTION

---

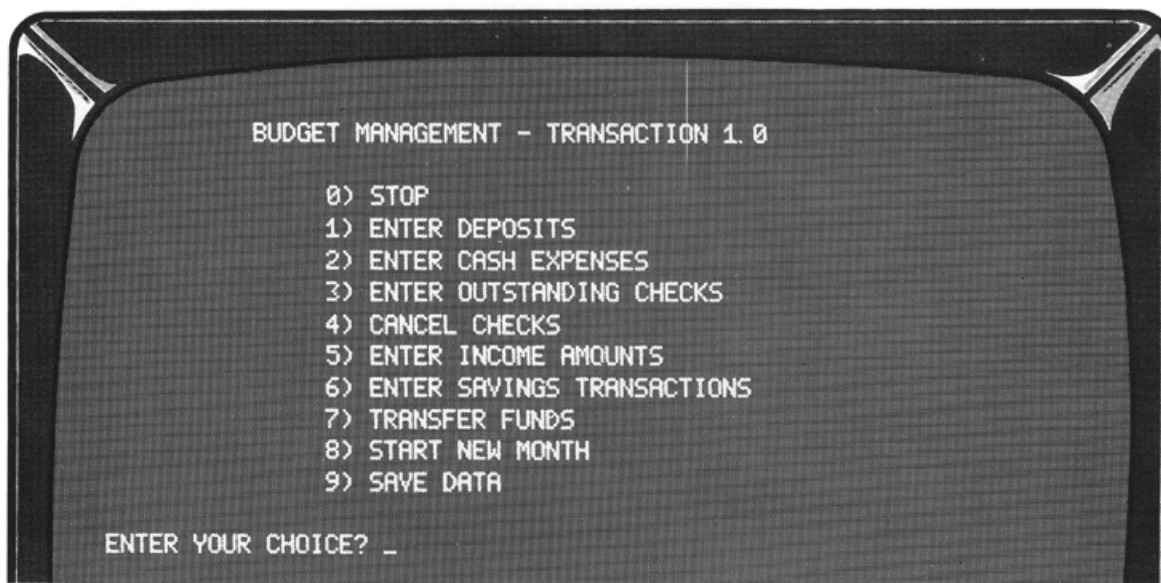
### How Does TRANS Work?

First, load the program and run it according to the Loading Instructions. The following will be displayed.

BUDGET MANAGEMENT - TRANSACTION 1.0

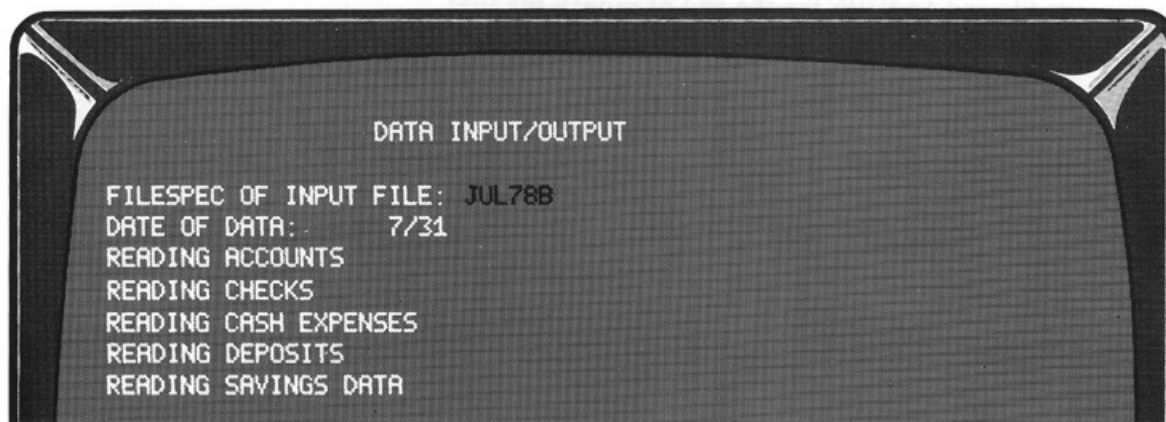
ENTER TODAY'S DATE (MM/DD/YY)? 07/31/78\_

Type in the date. The option list will then be displayed.



The first time you enter a choice 1-9 after you run the program, you have to load your account data into memory. Use the tape or disk with the most recent changes – the one you saved last.

Under DISK BASIC, the following will be displayed.





You should insert your most recent account diskette and enter the name of the most current account file.

Under LEVEL II, insert your most recent cassette and press **ENTER**. Be sure that your recorder is set ready to play.

LOAD DATA TAPE(PLAY), THEN PRESS ENTER?\_

When the data has been loaded, the option you chose will begin to execute. All the options except 8 are described below. Option 8 is described later on page 37.

### To Deposit To Checkbook

Option 1 allows you to deposit money in your checking account.

```
DEPOSITS

CHECKBOOK BALANCE          $124.78
DATE          DESCRIPTION          AMOUNT
? 07/25 ? PAYCHECK-$50          ? 352.99

CHECKBOOK BALANCE          $477.77
DATE          DESCRIPTION          AMOUNT
? END_
```

The current **CHECKBOOK BALANCE** is displayed before each deposit. You then enter the **DATE**, **DESCRIPTION**, and **AMOUNT** of your deposit. The new balance will be displayed and you will be allowed to enter deposits until you type **END** **ENTER** for the date.

If you press **ENTER** without typing a date, the Computer will assume that you want the **CURRENT DATE** you entered when the program first started.

The **DATE** you enter should be **MM/DD** format. Do not use the year for this entry. Month and day must each be two digits.

The **DESCRIPTION** should be no greater than 18 characters.

## TRANSACTION

### Outstanding Checks

Option 3 will allow you to record checks as you write them.

```

                                OUTSTANDING CHECKS

CHECKBOOK BALANCE             $477.77
  NO.    DATE    DESCRIPTION    AMOUNT    EXP. ACCT
  ? 101   ? 07/13 ? RENT         ? 140     ? 1200

CHECKBOOK BALANCE             $337.77
  NO.    DATE    DESCRIPTION    AMOUNT    EXP. ACCT
  ? 101   ? 07/20 ? ELECTRICITY  ? 25.01   ? 1400
    * CHECK EXISTS *

CHECKBOOK BALANCE             $337.77
  NO.    DATE    DESCRIPTION    AMOUNT    EXP. ACCT
  ? 102   ? 07/20 ? ELECTRICITY  ? 25.01   ? 1400

```

The current **CHECKBOOK BALANCE** is displayed before the entry of each check. The date is **MM/DD** format, and, again, if you press **ENTER** without typing a date, the Computer will use the date you typed at the beginning of the program.

If the **CHECK NUMBER** you recorded already exists, an error message **\* CHECK EXISTS \*** will be displayed.

**DESCRIPTIONS** should be no greater than 18 characters.

The **EXP. ACCT** is the account number of where you spent the money. The check amount will be added to the expense account you designate, and all main accounts will be corrected. For example, if you paid your rent by check and the rent account number is 550, use 550 as the **EXP. ACCT**. (You only need to enter the sub-account that you charge the check to. Main accounts will be figured automatically.)

You may also charge checks to your **SAVINGS ACCOUNT** number if you write a check to deposit into your savings, or use your **PETTY CASH** account number if you write a check to cash.

To void a check, use a **DESCRIPTION** of **VOID** and an **AMOUNT** of zero. You can use any sub-account you wish as the **EXP. ACCT**.



To reflect a service charge or any special charge, use a **CHECK NUMBER** below 100 and use a **MISCELLANEOUS ACCOUNT** number for the **EXP. ACCT**.

Type **END** **ENTER** for the check number when you are finished, and you will be returned to the option list.

## Cash Account

This account is used to record items you pay cash for and to keep track of your cash — your pocket money. Option 2 will allow you to enter a deposit or expenses on this account.

```

                                CASH EXPENSES

    PETTY CASH                      $75.50
    ENTER 'PETTY CASH' DEPOSIT (0 IF NONE)? 50

    PETTY CASH                      $125.50
    DATE      DESCRIPTION      AMOUNT    EXP. ACCT
    ? 07/01   ? LUNCH          ? 3.50    ? 1600

    PETTY CASH                      $122.00
    DATE      DESCRIPTION      AMOUNT    EXP. ACCT
    ? 07/17   ? SNACK          ? .76    ? 1600

    PETTY CASH                      $121.24
    DATE      DESCRIPTION      AMOUNT    EXP. ACCT
    ? END_
  
```

Any other time you get cash — for example, you deposit part of your paycheck and get cash for the rest — you should make a deposit to this account. If you have no deposit, type 0 **ENTER**.

Then, you will be asked to enter expenses. You should give the **DATE** in MM/DD format. If you press **ENTER** only, the **CURRENT DATE** you used before will be used.

The **DESCRIPTION** should be no greater than 18 characters.

The **EXP. ACCT** must be a valid account number, and it can have no sub-accounts. The amount of the cash expense will be added to this

## TRANSACTION

---

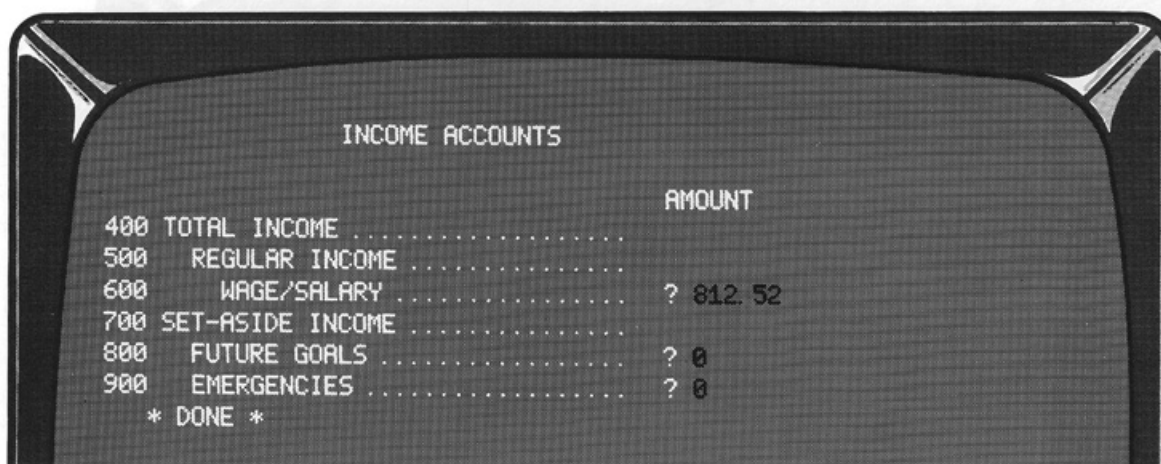
account automatically, and all main accounts will be adjusted automatically.

Type **END ENTER** for the **DATE** to return to the option list.

NOTE: If you wrote a check for cash, you can add it to your **PETTY CASH** account automatically by using the **PETTY CASH** account number for the **EXP. ACCT.** when you use option 3 to enter the check. Then, you don't have to use option 2.

### Income Accounts

Select option 5 to have all your income accounts displayed one at a time. You can enter an amount to add to each one. If you have nothing to add press **0**. Type **END ENTER** to return.



The screenshot shows a screen titled "INCOME ACCOUNTS". Below the title is a list of accounts with their corresponding amounts. The accounts are listed on the left, and the amounts are listed on the right. The amounts are displayed with a question mark and a dollar sign. The accounts and their amounts are:

	AMOUNT
400 TOTAL INCOME .....	
500 REGULAR INCOME .....	
600 WAGE/SALARY .....	? 812.52
700 SET-ASIDE INCOME .....	
800 FUTURE GOALS .....	? 0
900 EMERGENCIES .....	? 0
* DONE *	

NOTE: Entering an income amount will not add the amount to your checking or savings account. If you deposit this income into one of these accounts, you must choose another option to deposit it.

### Savings Accounts

The current **SAVINGS ACCOUNT** balance will be displayed before each savings transaction in option 6. Enter a negative amount by typing a minus sign, **-**, before the amount for withdrawals. Use positive amounts for deposits.

Press **ENTER** only to use the **CURRENT DATE** you entered previously.

The **DESCRIPTION** should be no greater than 18 characters.

---

Type **END** for the date to return to the option list.

SAVINGS TRANSACTIONS			
SAVINGS ACCOUNT		\$256.23	
DATE	DESCRIPTION	AMOUNT	
? 07/13	? CAR DOWNPAYMENT	? -200	
SAVINGS ACCOUNT		\$56.23	
DATE	DESCRIPTION	AMOUNT	
? 07/30	? PAYCHECK	? 480	
SAVINGS ACCOUNT		\$536.23	
DATE	DESCRIPTION	AMOUNT	
? END			

NOTE: If you wrote a check to deposit into savings, you can add it to your savings account automatically by using the **SAVINGS ACCOUNT** number for the **EXP. ACCT.** when you use option 3 to enter the check. Then, you don't have to use option 6.

## To Transfer Funds

Option 7 allows you to transfer money from any account to any other account (main account or sub-account).

After you enter the **ACCOUNT NUMBER** to which the money will be added that account's name and balance will be displayed.

Then, you can enter the **ACCOUNT NUMBER** from which the money will be taken. Its name and current balance will be displayed.

Finally, enter the **AMOUNT** to be transferred, and the action will take place. All main accounts will be corrected automatically to reflect any changes necessary.

Type **END ENTER** to return to the option list.

## TRANSACTION

---

TRANSFER FUNDS			
TO ACCOUNT NUMBER?	800	FUTURE GOALS	\$0.00
FROM ACCOUNT NUMBER?	300	PETTY CASH	\$121.24
AMOUNT?	20		
TO ACCOUNT NUMBER?	900	EMERGENCIES	\$0.00
FROM ACCOUNT NUMBER?	0		
AMOUNT?	25		
TO ACCOUNT NUMBER?	900	EMERGENCIES	\$25.00
FROM ACCOUNT NUMBER?	0		
AMOUNT?	0		
TO ACCOUNT NUMBER?	END		

NOTE: This option can be used to correct some errors. In the above example, when the **FROM ACCOUNT NUMBER** is zero, the **AMOUNT** is added to the **TO ACCOUNT NUMBER**. You should be careful using this option for changes like this.

### To Cancel Checks

Option 4 will probably be used whenever you get your monthly statement from the bank.

Simply type the **CHECK NUMBER** of the check you want to cancel. That check will be displayed on the next line.

If the check does not exist, the **\* CHECK NOT FOUND \*** message is displayed. If the check is already cancelled, the **\* CHECK ALREADY CANCELLED \*** message is displayed.

Type **END** **ENTER** to return to the option list.

## TRANSACTION

CANCELLED CHECKS

ENTER CHECK NUMBER? 103  
\* CHECK NOT FOUND \*

ENTER CHECK NUMBER? 101  
713 RENT \$140.00 1200

ENTER CHECK NUMBER? 101  
\* CHECK ALREADY CANCELLED \*

ENTER CHECK NUMBER? 102  
720 ELECTRICITY \$25.01 1400

ENTER CHECK NUMBER? END\_

NOTE: The **CURRENT DATE** will be used automatically as the **CANCEL DATE**.

### Finishing Up

You don't need to use all the options each time you run TRANS. Only request the options you need. When you finish, make sure you save your data before you stop. Use option 9, insert a diskette or blank cassette, and label the data according to the date of the transactions.

DATA INPUT/OUTPUT

FILESPEC OF OUTPUT FILE: DEC78

WRITING ACCOUNTS

WRITING CHECKS

WRITING CASH EXPENSES

WRITING DEPOSITS

WRITING SAVINGS DATA



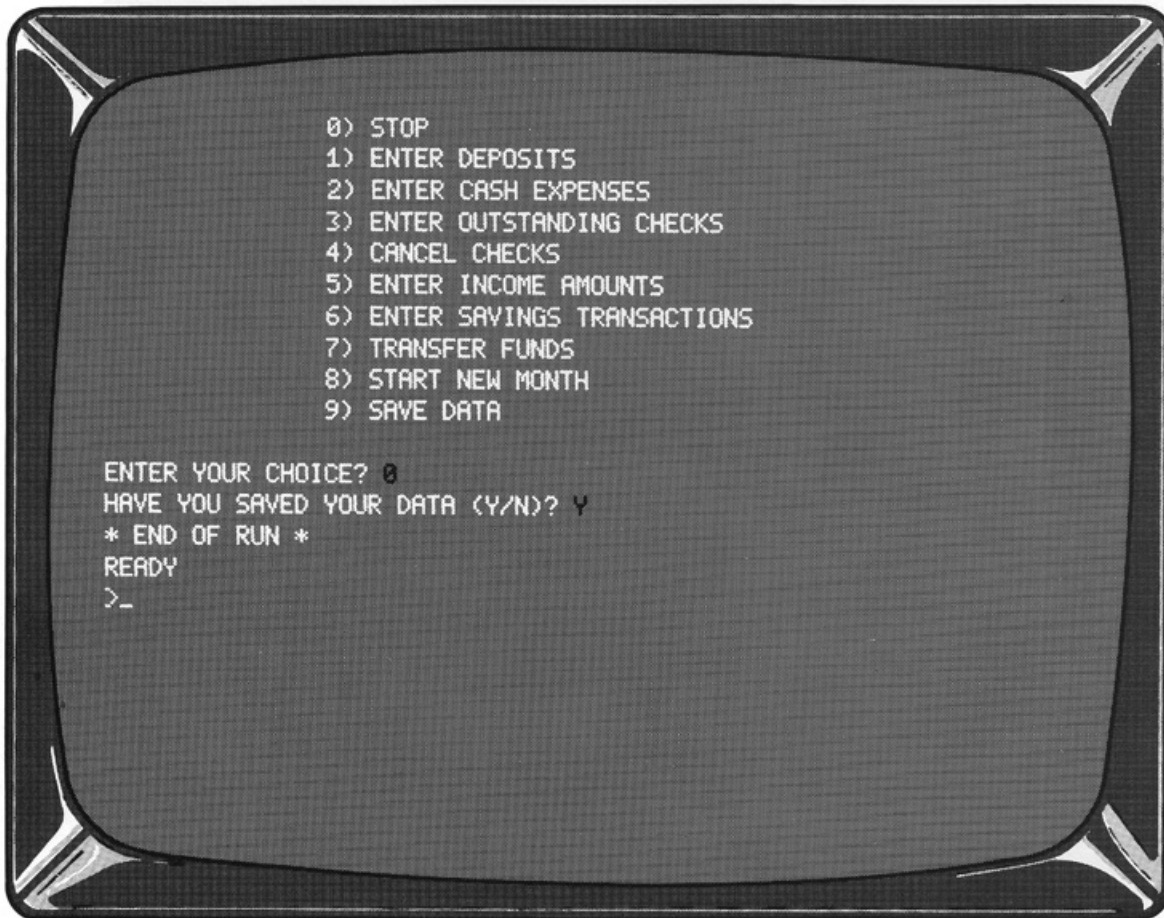
## TRANSACTION

---

LEVEL II BASIC users will see:

LOAD DATA TAPE(RECORD), THEN PRESS ENTER?\_

Remember, don't use a period as part of your filename on diskette.  
Then, use option 0 to end the session.



## What About Mistakes?

If you catch a mistake before you enter the transaction, you can either backspace to retype your correction or void the transaction by pressing **ENTER** before completing the information required for the transaction.

But if you have already completed the transaction and you do not discover the error until later, you can use the EDIT program to fix the mistake. EDIT allows you to examine and/or modify the data (accounts or transactions) created by INIT and TRANS. Use EDIT to check this data for errors and correct any that you might find.

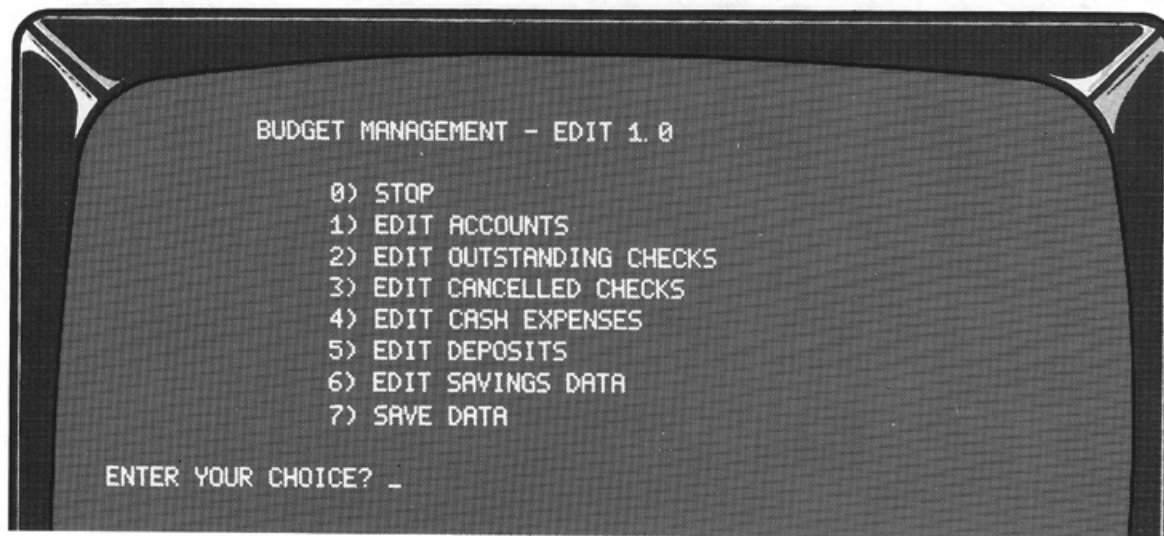
You should also use the EDIT program each time you run TRANS simply to check the changes you made for accuracy. This way you may be able to catch an error before it causes any problems.

First, run the EDIT program according to the Loading Instructions.

BUDGET MANAGEMENT - EDIT 1.0

ENTER TODAY'S DATE (MM/DD/YY)? 07/31/78\_

Enter the date as requested and the option list will be displayed.

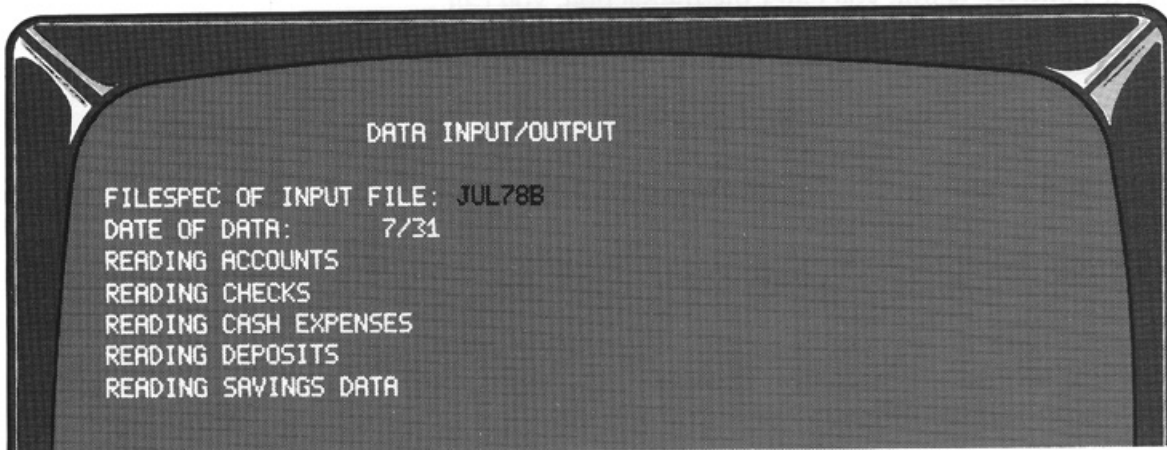


When you enter the number for the item you wish to check or change, you will be asked to load your accounts as before.

## EDIT

---

In DISK BASIC, you should **ENTER** the filespec for the data you want to verify.



In LEVEL II, load the correct cassette and press **ENTER**.

LOAD DATA TAPE(PLAY), THEN PRESS ENTER?\_

After the data is loaded, the first item for the option you chose will be automatically displayed. Press the down-arrow **↓** to display the next item. If the current item is the last one, a down-arrow will re-display it. Press the up-arrow key **↑** to display the previous item. If the current item is the first one, an up-arrow will re-display it.

Type an **E** to change the item currently on display. This will cause a question mark to appear under each part of that item allowing you to type information to replace that item with. Press **ENTER** only to leave it unchanged.

Type **X** to return to the option list.

If you want to enter data that is too large for that part of the item (an account number or a check number that does not exist) the **\* VALUE ERROR \*** message is displayed, and the current value will be used instead.



## To EDIT Accounts

Use option 1 to EDIT accounts.

ACCOUNT EDITING			
NO.	NAME	CURRENT	BUDGET
100	CHECKBOOK BALANCE	456.96	0.00 >v
200	SAVINGS ACCOUNT	163.75	0.00 >v
300	PETTY CASH	108.21	0.00 >v
400	TOTAL INCOME	752.99	817.50 >v
500	REGULAR INCOME	749.99	812.50 >v
600	WAGE/SALARY	749.99	812.50 >E
600	? -		

ACCOUNT EDITING			
NO.	NAME	CURRENT	BUDGET
100	CHECKBOOK BALANCE	456.96	0.00 >v
200	SAVINGS ACCOUNT	163.75	0.00 >v
300	PETTY CASH	108.21	0.00 >v
400	TOTAL INCOME	752.99	817.50 >v
500	REGULAR INCOME	749.99	812.50 >v
600	WAGE/SALARY	749.99	812.50 >E
600	PAYCHECKS	749.88	? -

Any changes in current or budget amounts of a sub-account will produce an automatic correction in all the main accounts affected by that sub-account.

## EDIT

### To EDIT Outstanding Checks

Use option 2.

OUTSTANDING CHECKS EDITING					
NO.	DATE	DESCRIPTION	AMOUNT	EXP.	ACCT
116	7/31	GAS CREDIT CARD	5.00	2800	>↓
118	7/31	GAS COMPANY	6.16	2000	>↓
119	7/31	TELEPHONE	34.42	2200	>E
119	7/31	PHONE	34.42	2200	
120	7/29	RENT	140.00	1600	>_

If a **CHECK NUMBER** is changed to any existing **CHECK NUMBER**, a **\* CHECK EXISTS \*** error is typed, and you will be asked to re-enter the check number. If you type the check's own number, a **\* CHECK EXISTS \*** message is also displayed. Press **ENTER** only to keep the same check number.

The **EXP. ACCT** must be a sub-account. It can't have any sub-accounts of its own. If the **EXP. ACCT** is changed to a main account or to an account that does not exist a **\* BAD EXPENSE ACCOUNT \*** message is displayed. You will then have to re-enter the account number.

If the **AMOUNT** or **EXP. ACCT** are changed to new values, the balances are corrected on that account and any other accounts that will be affected including main accounts.

### To EDIT Cancelled Checks

Use option 3.

CANCELLED CHECKS EDITING					
NO.	WRTN	CAN	DESCRIPTION	AMOUNT	EXP. ACCT
1	7/31	7/31	SERVICE CHARGE	1.50	3250 >↓
103	6/15	7/31	WATER	8.75	2100 >↓
109	7/31	7/31	GASOLINE CARD	5.00	2800 >↓
110	7/31	7/31	GAS COMPANY	5.12	2000 >↓
111	7/31	7/31	CAR PAYMENT	98.00	3000 >E
111	7/31	7/30	CAR PAYMENT	98.00	3000
112	7/31	7/31	BROWN SUIT	150.15	2400 >↓
113	7/31	7/31	WATER	8.30	2100 >↓
114	7/31	7/31	CAR PAYMENT	104.57	3000 >_

Beneath the **WRTN** column is the date the check was written.  
Beneath **CAN** is the date the check was cancelled.

If a check number is changed to any existing check number, a  
\* **CHECK EXISTS** \* message is displayed. You must re-enter a valid  
check number. If you type the check's own number, a \* **CHECK  
EXISTS** \* message is also displayed. Simply press **ENTER** to keep the  
same check number.

If the **EXP. ACCT** is changed to a main account or to an account  
that does not exist, a \* **BAD EXPENSE ACCOUNT** \* message occurs.  
If the **AMOUNT** or the **EXP. ACCT** number is changed, all the  
balances affected by the change will be corrected including main  
accounts.

## To EDIT Cash Expenses

Use option 4.

CASH EXPENSES EDITING			
DATE	DESCRIPTION	AMOUNT	EXP. ACCT
7/20	GROCERIES	50.00	2300 >↓
7/31	GROCERIES	50.00	2300 >↓
7/31	OIL PAINTS & BRUSHE	20.00	3200 >E
7/31	ART SUPPLIES	20.00	3200
7/31	ART SUPPLIES	20.00	3200 >_

If the **EXP. ACCT** is changed to a main account or to an account  
that does not exist, a \* **BAD EXPENSE ACCOUNT** \* message occurs.  
If the **AMOUNT** or **EXP. ACCT** number is changed, the balances  
affected by the change will be corrected including main accounts.

## To EDIT Deposits

Use option 5.

DEPOSITS EDITING		
DATE	DESCRIPTION	AMOUNT
7/14	PAYCHECK-\$70.99	304.00 >↓
7/28	PAYCHECK-\$50	334.99 >↓
7/28	PAYCHECK-\$50	334.99 >E
7/28	PAYCHECK LESS \$50	334.99
7/28	PAYCHECK LESS \$50	334.99 >_

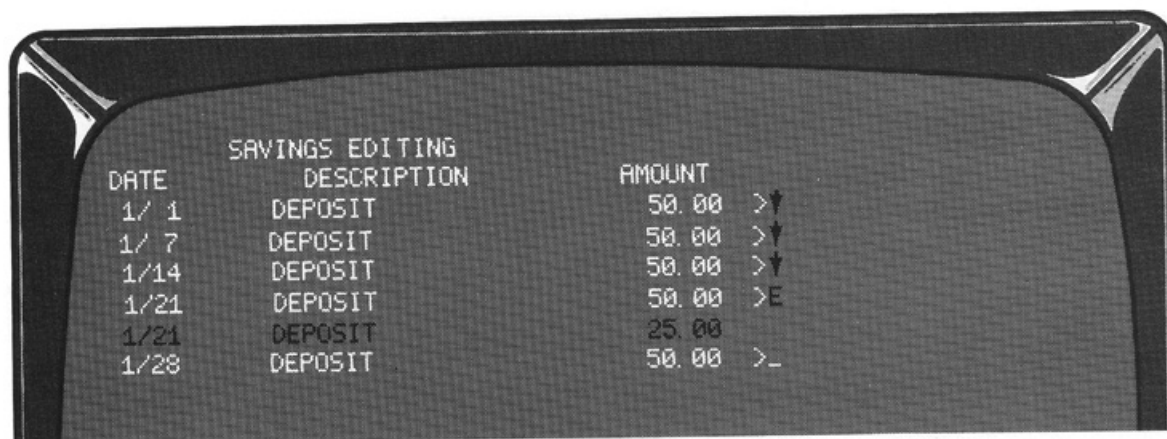
## EDIT

---

If the **AMOUNT** is changed, the **CHECKBOOK BALANCE** is corrected to reflect the new value.

### To EDIT Savings Account

Use option 6.



DATE	DESCRIPTION	AMOUNT
1/ 1	DEPOSIT	50.00 >
1/ 7	DEPOSIT	50.00 >
1/14	DEPOSIT	50.00 >
1/21	DEPOSIT	50.00 >
1/21	DEPOSIT	25.00 >
1/28	DEPOSIT	50.00 >

If the **AMOUNT** of a savings transaction is changed, the **SAVINGS ACCOUNT** balance is corrected to reflect the new value. The format of the savings data is the same as for **DEPOSIT** data above. Deposits to the **SAVINGS ACCOUNT** are positive numbers; withdrawals are negative. You must precede negative numbers with a minus sign, —.

Once you have checked all your files using the options described above, you can exit from the program using option 0.

**NOTE:** If you have made any changes, it will be necessary to save your data using option 9 before exiting from the program.

### What Happens At the End of the Month?

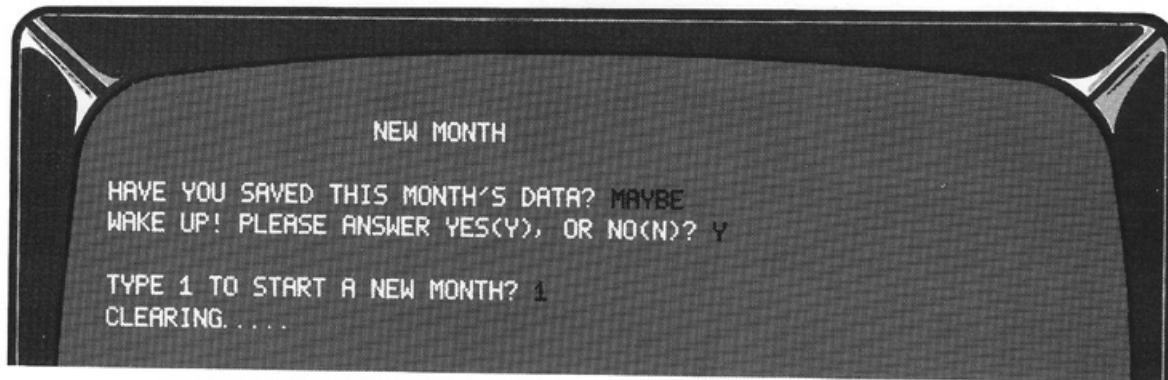
The tapes and diskettes you use to save your accounts on cannot hold an unlimited number of checks, savings transactions, expenses, etc. Each time you run TRANS you add to the file you previously had. Eventually, you must clear all these items and reset balances to make room for more data.

By convention, this is done after the end of each month. On the last day of the month, run TRANS and request any options you need to enter your last transactions for the month. Then, save this data as usual.

But the file you save this time reflects the entire previous month. So name or label the file after the month and year. (For example, JAN79, FEB79, etc.) This file will be saved for quite some time, so you might want to make a backup copy of it.

Then, the first week of the next month, or the first time you run TRANS during the next month, you load this permanent month-end file into memory.

BEFORE you do any normal transactions, select option 8 to start a new month. The following will be displayed to remind you to save last month's data.



Press **Y** if you already have it saved. Otherwise, press **N**, and you will have the chance to save it now.

Press **1** to the second question and items from last month will be cleared from memory. (The only record you will have of these items now is on the cassette or diskette for that month.) Enter anything but 1, and you will be returned to the option list, and no clearing will be done.

## STARTING A NEW MONTH

---

Starting a new month involves:

- Setting the **CURRENT** amount in all accounts to zero except for **CHECKBOOK BALANCE**, **SAVINGS ACCOUNT**, and **PETTY CASH**.
- All **CANCELLED CHECKS** are deleted, leaving only **OUTSTANDING CHECKS**.
- All **CASH EXPENSES**, **DEPOSITS**, and **SAVINGS** data are deleted.
- Budgeted amounts are also unchanged.

Once all this is done, you will be returned to the option list to enter your transactions as usual. Be sure to save your data when you get finished.



## How Do I View the Results?

You can use the fourth and last program in the package, REPORT, to print reports to the Video Display or to a printer.

You can choose to look at account data or at individual transaction items — outstanding or cancelled checks, cash expenses, checking deposits or savings transactions.

In the category you choose to look at, you can specify a range of dates, account numbers, check numbers, descriptions or amounts. For example, you can look at outstanding checks written for an amount over \$100.00.

Also, you can print cumulative reports for more than one month.

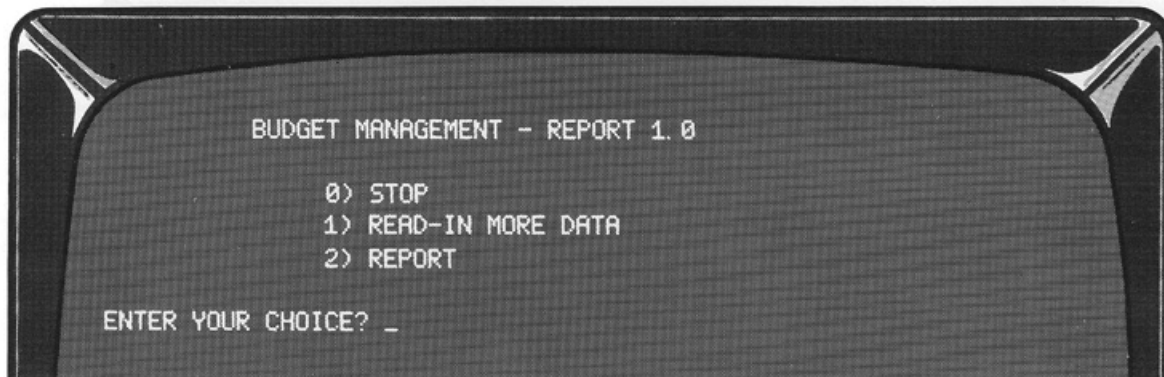
If you print the report (instead of displaying them), use a line printer, a Radio Shack Quick Printer, or a TRS-80 Screen Printer.

## To Print Monthly Reports

Run the REPORT program.

```
BUDGET MANAGEMENT - REPORT 1.0  
ENTER TODAY'S DATE (MM/DD/YY)? 07/31/78_
```

Enter the **CURRENT DATE** to display the first option list.



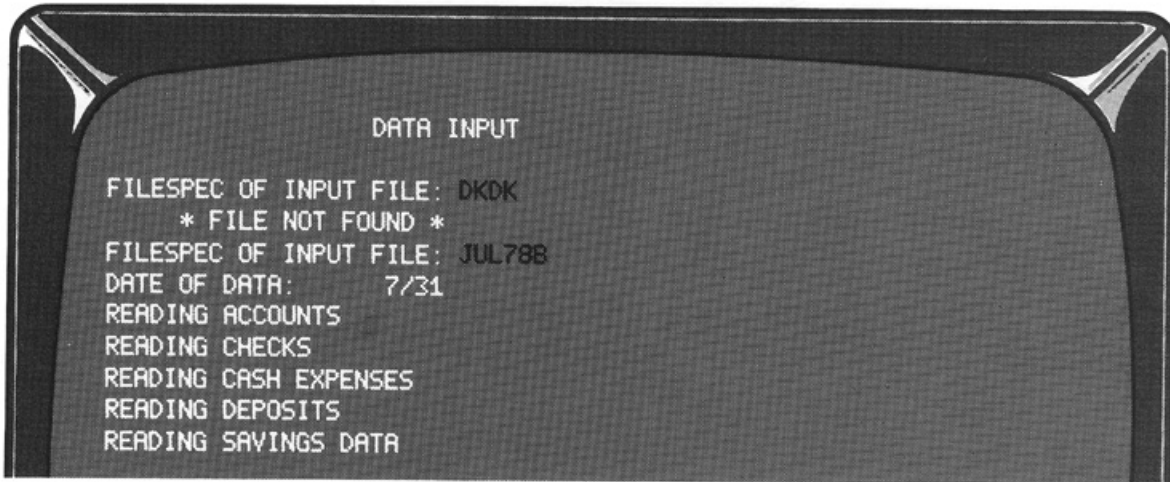
Type **2**. If there is no data in memory, you will be asked to load the file for which you want reports. (This can be any one data tape,

## REPORTS

---

permanent or temporary, and the reports will reflect the data on that tape.)

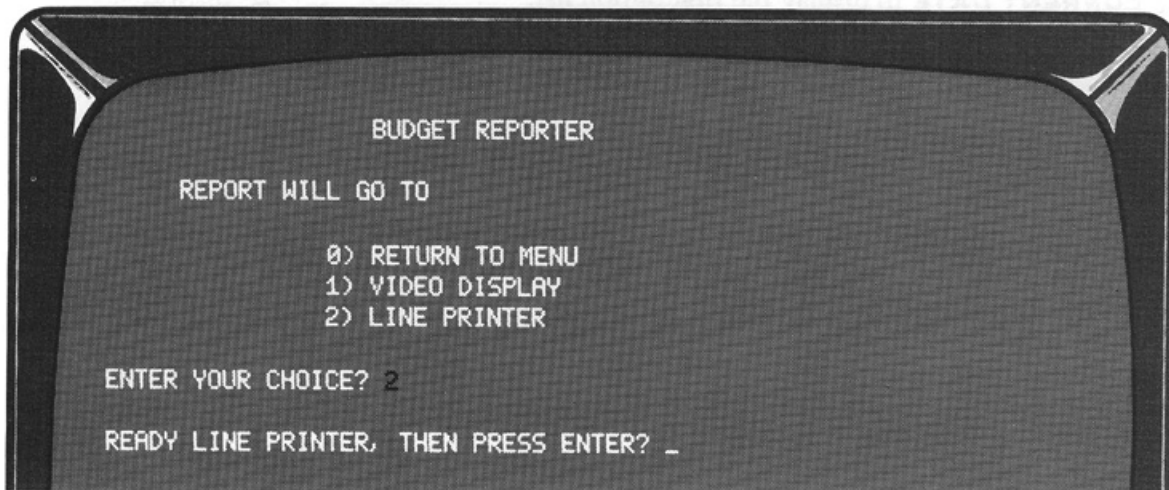
In DISK BASIC, the filespec will be requested.



In LEVEL II, the following prompt message is displayed instead:

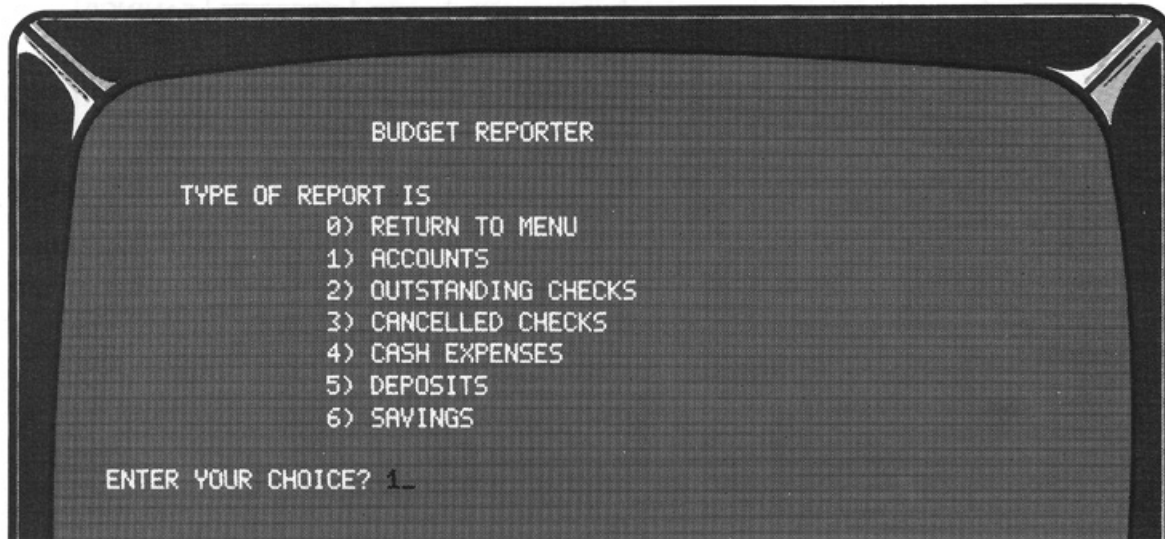
```
LOAD DATA TAPE<PLAY>, THEN PRESS ENTER?_
```

Use the cassette or diskette with the file you want reports for. When the file is loaded, the second option list will be displayed.



Enter your choice of where your reports are to be listed. If you use option 2, you must press **ENTER** after you ready your printer. When this is done, the third option list is displayed.





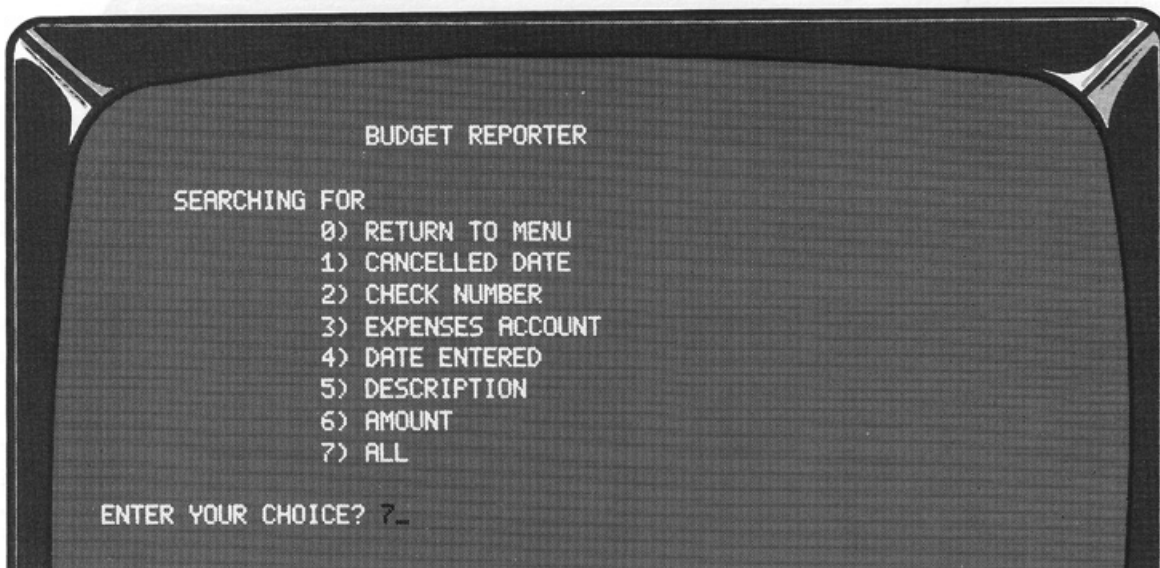
BUDGET REPORTER

TYPE OF REPORT IS

- 0) RETURN TO MENU
- 1) ACCOUNTS
- 2) OUTSTANDING CHECKS
- 3) CANCELLED CHECKS
- 4) CASH EXPENSES
- 5) DEPOSITS
- 6) SAVINGS

ENTER YOUR CHOICE? 1

Here, you can decide to look at accounts or look at specific transaction items. Enter your choice and the final option list will be displayed before the report is printed.



BUDGET REPORTER

SEARCHING FOR

- 0) RETURN TO MENU
- 1) CANCELLED DATE
- 2) CHECK NUMBER
- 3) EXPENSES ACCOUNT
- 4) DATE ENTERED
- 5) DESCRIPTION
- 6) AMOUNT
- 7) ALL

ENTER YOUR CHOICE? 7

You can enter a choice here to help you search for a specific item or several items. Your choice from this option list depends on the choice you made on the previous list. Not all of these are available for every account option.

## REPORTS

	ACCOUNT	OUTSTANDING	CANCELLED	CASH	DEPOSITS	SAVINGS
CANCEL DATE			V			
CHECK NO.		V	V			
ACCOUNT NO.	V	V	V	V		
DATE ENTERED		V	V	V	V	V
DESCRIPTION	V	V	V	V	V	V
AMOUNT	V	V	V	V	V	V
ALL	V	V	V	V	V	V

For example, if you are looking at accounts, the **CANCEL DATE** does not apply as a searching option.

Option 5 will allow you to select any single descriptive phrase to search for. On the other options, if you only want to find one specific item, enter the same value for **HIGH** and **LOW**. If the **LOW** value is greater than the **HIGH** value a \* **BAD RANGE** \* error occurs.

BUDGET REPORTER

SEARCHING FOR

- 0) RETURN TO MENU
- 1) CANCELLED DATE
- 2) CHECK NUMBER
- 3) EXPENSES ACCOUNT
- 4) DATE ENTERED
- 5) DESCRIPTION
- 6) AMOUNT
- 7) ALL

ENTER YOUR CHOICE? 6

<LOW AMOUNT>, <HIGH AMOUNT>      ? 75,300..

## REPORTS

When you enter a range value from this option list, only the data that falls into that range for the category you have chosen will be printed.

If there is no data in the range you specified or in the category you chose, a \* **NO ENTRIES** \* message is displayed.

At any time while a report is being printed, you can end it by pressing the @ key. You must then press **ENTER** to return to the option list.

### Note:

If you use a screen printer, you can print one screen at a time by pressing **ENTER** when the **PRESS ENTER TO CONTINUE?** message occurs.

Following are sample reports. The **PERCENT** column tells you what percentage of the budget has been used up to this point. The **REMAINING** column shows the difference between the **BUDGET** value and the **CURRENT** value.

---

### BUDGET MANAGEMENT REPORT

PAGE 1

#### \*\* DEPOSITS REPORT \*\*

DATE: 7/31/78

DATE =====	DESCRIPTION =====	AMOUNT =====
7/14	PAYCHECK-\$70.99 .....	304.00
7/28	PAYCHECK-\$50 .....	334.99

---

### BUDGET MANAGEMENT REPORT

PAGE 1

#### \*\* SAVINGS REPORT \*\*

DATE: 7/31/78

DATE =====	DESCRIPTION =====	AMOUNT =====
* NO ENTRIES *		

## REPORTS

BUDGET MANAGEMENT REPORT  
 \*\* OUTSTANDING CHECKS REPORT \*\*  
 DATE: 7/31/78

PAGE 1

NUMBER =====	DATE =====	DESCRIPTION =====	AMOUNT =====	EXPENSE =====
116	7/31	GASOLINE CHARGE CARD .....	5.00	2800
118	7/31	GAS COMPANY .....	6.16	2000
119	7/31	TELEPHONE .....	34.42	2200
120	7/29	RENT .....	140.00	1600

BUDGET MANAGEMENT REPORT  
 \*\* CANCELLED CHECKS REPORT \*\*  
 DATE: 7/31/78

PAGE 1

NUMBER =====	DATE =====	DESCRIPTION =====	AMOUNT =====	EXPENSE =====	CANCELLED =====
1	7/31	SERVICE CHARGE .....	1.50	3250	7/31
103	6/15	WATER .....	8.75	2100	7/31
109	7/31	GASOLINE CARD .....	5.00	2800	7/31
110	7/31	GAS COMPANY .....	5.12	2000	7/31
111	7/31	CAR PAYMENT .....	98.00	3000	7/31
112	7/31	BROWN SUIT .....	150.15	2400	7/31
113	7/31	WATER .....	8.30	2100	7/31
114	7/31	CAR PAYMENT .....	104.57	3000	7/31
115	7/31	ELECTRICITY .....	52.11	1900	7/31
117	7/31	CASH .....	40.00	2350	7/31

BUDGET MANAGEMENT REPORT  
 \*\* CASH EXPENSES REPORT \*\*  
 DATE: 7/31/78

PAGE 1

DATE =====	DESCRIPTION =====	AMOUNT =====	EXPENSE =====
7/20	GROCERIES .....	50.00	2300
7/31	GROCERIES .....	50.00	2300
7/31	OIL PAINTS & BRUSHES .....	20.00	3200

# REPORTS

## BUDGET MANAGEMENT REPORT

PAGE 1

\*\* ACCOUNTS REPORT \*\*

DATE: 7/31/78

ACCOUNT =====	NAME =====	CURRENT =====	BUDGET =====	PERCENT =====	REMAINING =====
100	CHECKBOOK BALANCE .....	456.96	0.00	0.00	0.00
200	SAVINGS ACCOUNT .....	163.75	0.00	0.00	0.00
300	PETTY CASH .....	108.21	0.00	0.00	0.00
400	TOTAL INCOME .....	752.99	817.50	92.11	64.51
500	REGULAR INCOME .....	749.99	812.50	92.31	62.51
600	WAGE/SALARY .....	749.99	812.50	92.31	62.51
700	VARIABLE INCOME .....	3.00	5.00	60.00	2.00
800	BONUS .....	0.00	0.00	0.00	0.00
900	INTEREST .....	3.00	5.00	60.00	2.00
1000	SAVINGS .....	3.00	5.00	60.00	2.00
1100	SET-ASIDE INCOME .....	0.00	0.00	0.00	0.00
1200	FUTURE GOALS .....	0.00	0.00	0.00	0.00
1300	EMERGENCIES .....	0.00	0.00	0.00	0.00
1400	TOTAL EXPENSES .....	779.08	695.07	112.09	84.01-
1500	REGULAR EXPENSES .....	779.08	695.07	112.09	84.01-
1600	RENT/MORTGAGE .....	140.00	140.00	100.00	0.00
1700	HOUSE MAINTENANCE .....	0.00	5.00	0.00	5.00
1800	UTILITIES .....	80.44	49.00	164.16	31.44-
1900	ELECTRICITY .....	52.11	25.00	208.44	27.11-
2000	NATURAL GAS .....	11.28	15.00	75.20	3.72
2100	WATER .....	17.05	9.00	189.44	8.05-
2200	TELEPHONE .....	34.42	25.00	137.68	9.42-
2300	FOOD & BEVERAGE .....	100.00	125.00	80.00	25.00
2350	MEALS OUT .....	40.00	50.00	80.00	10.00
2400	CLOTHING .....	150.15	5.00	3003.00	145.15-
2500	TRANSPORTATION .....	212.57	139.57	152.30	73.00-
2600	BUS .....	0.00	15.00	0.00	15.00
2700	CAR .....	212.57	124.57	170.64	88.00-
2800	GAS .....	10.00	15.00	66.67	5.00
2900	CAR MAINTENANCE .....	0.00	5.00	0.00	5.00
3000	CAR PAYMENTS .....	202.57	104.57	193.72	98.00-
3100	MEDICAL .....	0.00	5.00	0.00	5.00
3200	MISCELLANEOUS .....	20.00	150.00	13.33	130.00
3250	SERVICE CHARGE .....	1.50	1.50	100.00	0.00
3300	SEASONAL EXPENSES .....	0.00	0.00	0.00	0.00
3400	EDUCATION .....	0.00	0.00	0.00	0.00
3500	TRAVEL .....	0.00	0.00	0.00	0.00

## REPORTS

If you use a screen printer:

BUDGET MANAGEMENT: ACCOUNTS		PAGE 1 DATE 7/31/78		
NO.	NAME	CURRENT	BUDGET	REMAINING
100	CHECKBOOK BALANCE .....	456.96	0.00	0.00
200	SAVINGS ACCOUNT .....	163.75	0.00	0.00
300	PETTY CASH .....	108.21	0.00	0.00
400	TOTAL INCOME .....	752.99	817.50	64.51
500	REGULAR INCOME .....	749.99	812.50	62.51
600	WAGE/SALARY .....	749.99	812.50	62.51
700	VARIABLE INCOME .....	3.00	5.00	2.00
800	BONUS .....	0.00	0.00	0.00
900	INTEREST .....	3.00	5.00	2.00
1000	SAVINGS .....	3.00	5.00	2.00
1100	SET-ASIDE INCOME .....	0.00	0.00	0.00
1200	FUTURE GOALS .....	0.00	0.00	0.00
1300	EMERGENCIES .....	0.00	0.00	0.00

PRESS ENTER TO CONTINUE? \_

BUDGET MANAGEMENT: ACCOUNTS		PAGE 2 DATE 7/31/78		
NO.	NAME	CURRENT	BUDGET	REMAINING
1400	TOTAL EXPENSES .....	779.08	695.07	84.01-
1500	REGULAR EXPENSES .....	779.08	695.07	84.01-
1600	RENT/MORTGAGE .....	140.00	140.00	0.00
1700	HOUSE MAINTENANCE ..	0.00	5.00	5.00
1800	UTILITIES .....	80.44	49.00	31.44-
1900	ELECTRICITY .....	52.11	25.00	27.11-
2000	NATURAL GAS .....	11.28	15.00	3.72
2100	WATER .....	17.05	9.00	8.05-
2200	TELEPHONE .....	34.42	25.00	9.42-
2300	FOOD & BEVERAGE ....	100.00	125.00	25.00
2350	MEALS OUT .....	40.00	50.00	10.00
2400	CLOTHING .....	150.15	5.00	145.15-
2500	TRANSPORTATION.....	212.57	139.57	73.00-

PRESS ENTER TO CONTINUE? \_

BUDGET MANAGEMENT: ACCOUNTS		PAGE 3 DATE 7/31/78		
NO.	NAME	CURRENT	BUDGET	REMAINING
2600	BUS .....	0.00	15.00	15.00
2700	CAR .....	212.57	124.57	88.00-
2800	GAS .....	10.00	15.00	5.00
2900	CAR MAINTENANCE .....	0.00	5.00	5.00
3000	CAR PAYMENTS ...	202.57	104.57	98.00-
3100	MEDICAL .....	0.00	5.00	5.00
3200	MISCELLANEOUS .....	20.00	150.00	130.00
3250	SERVICE CHARGE .....	1.50	1.50	0.00
3300	SEASONAL EXPENSES ....	0.00	0.00	0.00
3400	EDUCATION .....	0.00	0.00	0.00
3500	TRAVEL .....	0.00	0.00	0.00

PRESS ENTER TO CONTINUE? \_

## REPORTS

You can cancel reports by pressing the @ key. Then answer the question by pressing the **ENTER** key.

BUDGET MANAGEMENT: CANCELLED CHECKS				PAGE 1	DATE 7/31/78
NO.	WRTN	CAN	DESCRIPTION	AMOUNT	EXP. ACCT
1	7/31	7/31	SERVICE CHARGE .....	1.50	3250
103	6/15	7/31	WATER .....	8.75	2100
109	7/31	7/31	GASOLINE CARD .....	5.00	2800
110	7/31	7/31	GAS COMPANY .....	5.12	2000
111	7/31	7/31	CAR PAYMENT .....	98.00	3000

\* ABORTED \*

PRESS ENTER TO CONTINUE? \_

## REPORTS

---

### Printing Cumulative Reports

You can also use the REPORT program to print reports for more than one month, even though each month is on a separate tape or diskette file.

When you first run the program, use option 1 on the first option list. After you read in the first tape or diskette file, you will be returned to the option list.

Continue to use option 1 until you have read in the last file you want to use. Then, use option 2.

The rest of the instructions are the same as for the single reports.

NOTE: After the first cassette or diskette has been read, only accounts that match those on it will be read. If the account is not on the first file, it will be ignored on later cassette or diskette files. The **CURRENT** and **BUDGET** values are added.

Thus, if you have added accounts at some time after you first created them, the latest file with the most accounts on it should be read in first.

This is why you can't run cumulative reports if you have renumbered your accounts since you began. All of your old account numbers will not match the new ones, and one set will be ignored.

### Tips On Reading the Reports

Pay close attention to the **REMAINING** column. You can use it to guide changes in your spending habits. If you have a lot left over in one category that you budgeted but did not spend, this tells you that the **BUDGET** amount is not realistic. If you have a negative amount, you spent more than you budgeted.

You may change the **BUDGET** amount using the EDIT program. You can change your spending habits using your will power.

If you have trouble seeing exactly where your money is going, you may want to create more sub-accounts or insert new accounts using the INIT program.

You can judge all these things better after you've used the program for a few months. So experiment and enjoy yourself using these programs; they are at your service. Try playing with some test data for a few days to get a "feel" for the package.

Happy Budgeting!

---



# APPENDIX

Worksheets .....	51
Generating Your Own Accounts .....	55
Using the Radio Shack Quick Printer .....	59
Error Messages .....	61
Variable Usage .....	65
Memory Usage .....	67
Program Listings .....	69



**(Worksheet)**

[illegible]

**(Worksheet)**

[illegible]

**(Worksheet)**

[illegible]



## Generating Your Own Account List

If you wish to generate your own account list, you must create **DATA** statements to replace those found in the INIT program. You need to have a little programming experience to do this because you will have to modify the program.

It would be a good idea to save a copy of the original version even after your changes are complete.

First **CLOAD** the INIT program (the only one you have to change). Then, use the **DELETE** command in BASIC to delete lines 10000-10090 as follows.

```
DELETE 10000-10090
```

This deletes the default account list. Now, you must replace it with your own accounts from the Worksheet you filled out. You need to use the **ACCOUNT NAME** and **LEVEL** for now.

The first account **MUST** be **CHECKBOOK BALANCE** with a Level of zero.

The second account **MUST** be **SAVINGS ACCOUNT** with a Level of zero.

The third account **MUST** be **PETTY CASH** with a Level of zero.

The fourth account **MUST** be **TOTAL INCOME** with a Level of zero. This account must have at least one sub-account beneath it.

After your income accounts you **MUST** have the **TOTAL EXPENSES** account with a Level of zero. This account must also have at least one sub-account beneath it.

You may then add as many zero level accounts as you wish. Each zero level account starts an independent block of accounts.

The following example sets up a couple's income and expenses separately.

# Worksheet

[illegible]



Start entering your accounts as **DATA** statements at line 100000. Each line should be numbered in increments of 10. Also, each line should begin with the word **DATA** followed by the **ACCOUNT NAME**, a comma, the account's Level, a comma, the next account's name, a comma, that account's Level, and so on.

The **DATA** statements would look like this for the example above:

```
100000 DATA CHECKBOOK BALANCE, 0, SAVINGS ACCOUNT, 0, PETTY CASH, 0
100100 DATA TOTAL INCOME, 0, HUSBAND'S PAY, 1, INTEREST INCOME, 1
100200 DATA WIFE'S INCOME, 0, WIFE'S PAY, 1, INTEREST INCOME, 1
100300 DATA TOTAL EXPENSES, 0, HOUSE PAYMENTS, 1, UTILITIES, 1
100400 DATA WIFE'S EXPENSES, 0, CAR PAYMENTS, 1, TELEPHONE, 1
100500 DATA JOINT EXPENSES, 0, COOKIES, 1
100600 DATA FOOD & BEVERAGE, 1, " ", -1
```

**BEWARE!** Don't use commas or colons as part of **ACCOUNT NAMES**.

After the last account is a single space enclosed in quotes, with a level of minus one. This must **ALWAYS** be the last two pieces of **DATA**.

Save this new **INIT** program on cassette or diskette when you are done so you can run it.

The new version of **INIT** will use your accounts instead of the ones originally programmed. You can press **Y** to use each account. Then, you will be asked a **CURRENT** and **BUDGET** amount for each of your accounts just as in the original program.

If you receive a **\* BAD LEVEL \*** or a **\* FATAL ERROR ABORTED \*** message, it means the program finds your indentation levels inconsistent. Go back through your **DATA** statements to see if you can find the error and correct it.

Finally, use option 4 of the **INIT** program as usual to save your account structure.

You can renumber it, or add to it at any time.



### Using the Quick Printer

If you use a Radio Shack Quick Printer for your reports and you'd like to keep your printouts from overflowing to 2 lines, you can print the Video Display on the Quick Printer.

Simply change all **PRINTS** to **LPRINTS** in lines 3510-3520, lines 6010-6060 and in line 1010 and line 2365 of the REPORT program. Then, ask for Video Output (option 1 in the second option list of the REPORT program). The reports will go to the Quick Printer.



## **Error Messages**

### **UNEXPECTED ERROR #nn IN LINE xxxx-RUN ABORTED**

Error nn occurred while opening, closing, reading, or writing to disk or tape.

If this error occurs because you used an invalid TRSDOS filespec when you tried to load or save your data files, you can type **RESUME 220 ENTER** to try again.

### **\*FILE NOT FOUND\***

Specified input file does not exist.

### **\*INPUT DATA EXCEEDS MEMORY ALLOCATION\***

Too much data in the data file. Start a new file.

### **\*BAD LEVEL\***

Level is inconsistent with previous level or is greater than five.

### **\*FATAL ERROR\***

The program was aborted due to a serious error.

### **\*BAD ACCOUNT NUMBER SEQUENCE\***

Invalid renumbering request.

### **\*NO ROOM TO INSERT\***

All memory allocated to accounts has been used and no room exists for new account insertion. Start a new file.

### **\*BAD ACCOUNT NUMBER\***

The new account number for insertion exists or is too large or small.

### **\*NO CHECKS\***

No outstanding checks exist so no cancelling can be done.

### **\*CHECK EXISTS\***

Attempt to enter a check with the same check number as a previously entered check.

### **\*BAD EXPENSE ACCOUNT\***

The specified account was a main-account or does not exist.

### **\*CHECK NOT FOUND\***

No check with specified check number exists.

### **\*CHECK ALREADY CANCELLED\***

Attempt to re-cancel a previously cancelled check.

---

## APPENDIX

---

**\*CASH EXPENSES FULL\***

All allocated memory for cash expenses has been used and no new entries may be made. Start a new file.

**\*DEPOSITS FULL\***

All allocated memory for deposits has been used and no new entries may be made. Start a new file.

**\*SAVINGS DATA FULL\***

All allocated memory for savings transactions has been used and no new entries may be made. Start a new file.

**\*CHECK FULL\***

All allocated memory for checks has been used and no new entries may be made. Start a new file.

**\*NO DATA\***

The selected data category has no entries, and therefore no editing can be performed.

**\*BAD VALUE\***

The entered check number, date, or account number was too large (greater than 32767).

**\*ABORTED\***

The at-symbol (@) was depressed during a print-out to cancel the report being generated.

**\*BAD REQUEST\***

The searching request was invalid for the previously selected data category.

**\*NO ENTRIES\***

No data exists in the specified data category or no data was found in the designated range during a search.

If any **FULL** error messages occur, you have to use option 8 in the TRANS program to **START A NEW MONTH**, even if it is not yet the end of the month. You can end your new file on schedule at month-end. Then, you will have two permanent files for that month, and you will need to use both of them as input to REPORT.

**Note:** When a month is split into two parts, you must EDIT the "budget" amounts and change them all to half their value. (If this is not done, the budget amounts are added together at report time. This would double your "budget" amounts.)

Or you can continue with the new file past the month-end until it gets full also. You will start a new file whenever this happens from now on. If you do this, you will no longer be on a monthly schedule. This might turn out to be inconvenient, since many of your budgeted items occur on a monthly basis.



## APPENDIX

---

### Variable Usage

M1	Maximum number of accounts allowed
M2	Maximum number of checks
M3	Maximum number of cash expenses
M4	Maximum number of deposits
M5	Maximum number of savings transactions
K1	Current number of accounts in memory
K2	Current number of checks in memory
K3	Current number of cash expenses in memory
K4	Current number of deposits in memory
K5	Current number of savings transactions in memory
A1%	Account number (negative if sub-account)
A2\$	Account name (18 characters)
A3#	Current account amount
A4#	Budget (predicted) account amount
A6%	Reference/level value (See below.)
C1%	Check number
C2%	Date written
C3\$	Description (18 characters)
C4	Amount of check
C5%	Expense account index (See below.)
C6%	Date cancelled (zero if not cancelled)
E1%	Date of cash expense
E2\$	Description (18 characters)
E3	Amount of cash expense
E4%	Expense account index (See below.)
D1%	Date of deposit
D2\$	Description (18 characters)
D3	Amount of deposit
S1%	Date of savings transaction
S2\$	Description (18 characters)
S3	Amount (positive for deposit; negative for withdrawal)
DT	Today's date
F	Data file flag: F=0 if not data file in memory F=1 if data file is not in memory
X	String for data delimiter-set to a text comma (",")
AD\$	Advance cursor right
EL\$	Erase to end of line
CU\$	Move cursor up
BL\$	Move to beginning of line
CD\$	Move cursor down
J	Current menu selection
D	Date of data file
DK	Disk flag: DK=0 if LEVEL II BASIC Computer DK=1 if DISK BASIC Computer

**Reference/Level Value**

$$\text{Value} = \text{Reference Index} * 10 + \text{Level}$$

The Level here is the same as that on the Worksheet. The Reference points to the Index on the main-account or is equal to zero if there is no main-account.

**Expense Account Index**

Instead of saving the actual expense account, a pointer is stored as an Index to the desired Expense Account.

## APPENDIX

---

### Memory Usage

The information below may be used by an experienced programmer to take advantage of larger RAM memory than 16K. The Budget Management Package is written for:

61 Accounts (The 61st Account is always an END-OF-ACCOUNT marker.)  
50 Checks  
20 Deposits  
25 Cash Expenses  
20 Savings Transactions

The following calculations must be performed to assure an **OUT OF MEMORY** error does not occur.

ITEM	BYTES USED
Program text	8324
Program string space	432
Program stack space and variables (overhead)	860
Account (41 each * 61) =	2501
Checks (33 each * 50) =	1650
Cash expenses (29 each * 25) =	725
Deposits (27 each * 20) =	540
Savings (27 each * 20) =	+ 540
Total =	15572

M1 = one plus number of accounts desired  
M2 = number of checks desired  
M3 = number of cash expenses desired  
M4 = number of deposits desired  
M5 = number of savings transactions desired

String Space =  $432 + 18 * (M1 + M2 + M3 + M4 + M5)$

The result replaces 3600 in line 100 of each program as the amount of string space.

Memory Available  $> 9616 + M1 * 41 + M2 * 33 + M3 * 29 + M4 * 27 + M5 * 27$

If the memory you have in your system is less than the above, an **OUT OF MEMORY** error will occur when you try to run the program.

You should change line 110 in each program to reflect the correct values of variables M1 through M5.

## **Program Listings**

<b>INIT .....</b>	<b>69-72</b>
<b>TRANS .....</b>	<b>73-76</b>
<b>EDIT .....</b>	<b>77-81</b>
<b>REPORT .....</b>	<b>82-86</b>



```
10 / BUDGET MANAGEMENT - INITIALIZATION 1.0
20 / COPYRIGHT (C) 1979 TANDY CORP.
100 CLEAR(3600):DEFINTI-N:DEFSTRX
110 M1=61:M2=50:M3=25:M4=20:M5=M4
120 X="":AD$=CHR$(25):EL$=CHR$(30):CU$=CHR$(27):BL$=CHR$(29)
130 DIMA1%(M1),A2$(M1),A3$(M1),A4$(M1),A6%(M1)
140 DIMC1%(M2),C2$(M2),C3$(M2),C4$(M2),C5$(M2),C6$(M2)
150 DIME1%(M3),E2$(M3),E3$(M3),E4$(M3)
160 DIMD1%(M4),D2$(M4),D3$(M4)
170 DIMS1%(M5),S2$(M5),S3$(M5)
190 ONERRORGOTO200:DK=1:I=INSTR("A","A"):GOTO210
200 DK=0:RESUME210
210 ONERRORGOTO0
220 CLS:PRINT:PRINTTAB(10)"BUDGET MANAGEMENT - INITIALIZATION 1.0":PRINT
230 IFDT<10100PRINT"ENTER TODAY'S DATE (MM/DD/YY)":X%=1:GOSUB900:DT=VAL(DT$):X%
  =0:GOTO220
240 PRINTTAB(15)"0) STOP
    "TAB(15)"1) CREATE ACCOUNTS
    "TAB(15)"2) INSERT ACCOUNTS
    "TAB(15)"3) RENUMBER ACCOUNTS
    "TAB(15)"4) SAVE DATA
260 PRINT:INPUT"ENTER YOUR CHOICE";J
270 IFJ=0THENC$="":PRINT:PRINT"HAVE YOU SAVED YOUR DATA (Y/N)":GOSUB300:IFYN=0T
  HEN220ELSEE=99:GOSUB5000
272 IF(J-1)*(J-2)*(J-3)*(J-4)<>0THEN220
275 IFJ<>1ANDF=0THENGOSUB9500
280 ONJGOSUB1000,2000,3000,9000:GOTO220
300 YN=-1:GOSUB700:PRINT:IFB$=""THENB$=C$
305 C$=LEFT$(B$,1):IFC$="Y"THENYN=1:RETURNELSEIFC$="N"THENYN=0:RETURN
310 PRINT"WAKE UP! PLEASE ANSWER YES(Y), OR NO(N)":IFF1=1THENRETURN:ELSE300
320 FORDY=1TO1000:NEXT:RETURN
340 LL=INT(A6%(I)-10*INT(A6%(I)/10)):RETURN
350 CLS:IFJ=4THENPRINTTAB(18)"DATA INPUT/OUTPUT":RETURN
355 PRINTTAB(18)"ACCOUNT ":ONJGOTO361,362,363
361 PRINT"INITIALIZATION":RETURN
362 PRINT"INSERTION":RETURN
363 PRINT"RENUMBERING":RETURN
370 PRINTUSING"##### ";ABS(N):PRINTSTRING$(2*L," ");A$:TAB(34):RETURN
380 GOSUB700:A3$(J)=VAL(B$):PRINTBL$:STRING$(48,AD$):EL$:GOSUB700:A4$(J)=VAL(B$
  ):PRINT
390 IFJ>1THENA1%(J)=-1*A1%(J):RETURNELSERETURN
500 R=0:FORI=K1-1TO4STEP-1:GOSUB340:IFLL+1=LTHENR=I:GOTO510:ELSENEXTI
510 IFR<>0THENA6%(K1)=A6%(K1)+R*10:RETURN
520 E=3:GOSUB5000
530 E=5:GOSUB5000:STOP
650 K=INT(A6%(K)/10):IFK=0THENRETURN
660 A3$(K)=A3$(K)+A3$(M):A4$(K)=A4$(K)+A4$(M):GOTO650
700 PRINT"? ";
750 B$="":L=0:PRINTCHR$(14);
760 A$=INKEY$:IFA$=""THEN760
770 IFA$=CHR$(13)THENPRINTCHR$(15):L=L+1:RETURN
780 IFA$=CHR$(8)THENIFB$<>""THENPRINTA$:B$=LEFT$(B$,LEN(B$)-1):GOTO760ELSE760
790 IFA$=""THENL=L+1:PRINT " ":GOTO760
```

```

800 IFA$="" THEN PRINT STRING$(LEN(B$)+2*L, CHR$(8)); EL$; :GOTO750
810 IFA$<" " OR A$>"Z" THEN 760 ELSE PRINT A$; :B$=B$+A$;GOTO760
900 GOSUB700: IF LEN(B$)<5+3*X% THEN 930 ELSE D1$=LEFT$(B$, 2):D2$=MID$(B$, 4, 2):D3$=RIGHT$(B$, 2): IF VAL(D1$)<10 OR VAL(D1$)>12 THEN 930
905 IF VAL(D2$)<10 OR VAL(D2$)>31 THEN 930 ELSE IF X%=1 THEN IF VAL(D3$)<0 THEN 930
910 DT$=D1$+D2$: IF X%=1 THEN DT$=DT$+D3$: RETURN ELSE RETURN
930 DT$=" ": RETURN
950 PRINT USING "###/###"; INT(D/100); D-100*INT(D/100); : RETURN
1000 GOSUB350: PRINT TAB(32)"WANTED(Y/N)": N=0: K1=N: F1=1: RESTORE
1010 READ A$, L
1020 IFL<0 THEN 1100
1025 IFL>5 THEN E=3: GOSUB5000: PRINT A$, L: GOTO530
1030 K1=K1+1: IF K1+1>M1 THEN E=2: GOSUB5000: GOSUB320: RETURN
1040 A2$(K1)=A$: IFL=0 THEN N=(1+INT(N/100))*100: ELSE N=N+10
1050 GOSUB370: IFL=0 THEN PRINT: GOTO1070
1060 PRINT BL$; STRING$(34, AD$); EL$; : F1=1: L1=L: GOSUB300: L=L1: IF YN=1 THEN 1070
1065 IF YN=0 THEN K1=K1-1: GOTO1090: ELSE PRINT CU$; : GOTO1060
1070 A1$(K1)=N: A6$(K1)=L
1085 IFL=0 THEN 1010 ELSE GOSUB500: GOTO1010
1090 READ A$, LL: IFL>L THEN N=N+10: GOTO1090 ELSE L=LL: GOTO1020
1100 K=K1+1: A1$(K)=32001: A2$(K)="END-OF-ACCOUNTS"
1110 GOSUB350: PRINT TAB(34)"CURRENT" TAB(48)"BUDGET"
1120 FOR J=1 TO K1: N=A1$(J): A$=A2$(J): I=J: GOSUB340: L=LL: I=J+1: GOSUB340: GOSUB370
1130 IF (L=0 OR L<LL) AND J>3 THEN PRINT: GOTO1190
1140 GOSUB380: K=J: M=J: GOSUB650
1190 NEXT J: F=1: F1=0: RETURN
2000 GOSUB350: CD$=CHR$(26)
2010 IF K1+2>M1 OR K1=0 THEN E=7: GOSUB5000: GOSUB320: RETURN
2020 PRINT: PRINT "ENTER NEW ACCOUNT NUMBER"; : GOSUB700: PRINT: NN=VAL(B$): IF B$="END" RETURN
2030 M=0: FOR I=4 TO K1: IF NN>ABS(A1$(I)) AND NN<ABS(A1$(I+1)) THEN M=I+1: NEXT ELSE NEXT
2040 IF M=0 THEN E=8: GOSUB5000: GOTO2020
2045 PRINT STRING$(5, CHR$(13))+STRING$(4, CU$);
2050 N=A1$(M-1): A$=A2$(M-1): I=M-1: GOSUB340: L=LL: GOSUB370: PRINT "CURRENT" TAB(48)"BUDGET"
2060 PRINT: N=A1$(M): A$=A2$(M): I=M: GOSUB340: L=LL: GOSUB370: PRINT BL$; CU$;
2070 PRINT USING "##### "; NN; : GOSUB750: IF B$="" THEN PRINT BL$; CD$; CD$; EL$; : GOTO2020
2080 I=M-1: GOSUB340: IFL=LL OR L=LL+1 THEN 2100
2090 E=3: GOSUB2500: PRINT EL$; : GOTO2070
2100 FOR I=K1+2 TO M STEP -1: A1$(I)=A1$(I-1): A2$(I)=A2$(I-1): A3$(I)=A3$(I-1): A4$(I)=A4$(I-1): A6$(I)=A6$(I-1): NEXT
2110 KK=K1+1: A1$(M)=NN: A2$(M)=B$: A3$(M)=0: A4$(M)=0: A6$(M)=L: GOSUB2600
2140 PRINT BL$; STRING$(34, AD$); EL$; : GOSUB700: A3$(M)=VAL(B$)
2150 PRINT BL$; STRING$(48, AD$); EL$; : GOSUB700: A4$(M)=VAL(B$)
2160 I=M: GOSUB340: L=LL: K1=M: GOSUB500: K=M: GOSUB650: K1=KK
2163 I=M-1: GOSUB340: IFL>LL THEN A1$(M-1)=ABS(A1$(M-1))
2167 I=M+1: GOSUB340: IFL>LL THEN A1$(M)=-1*ABS(A1$(M))
2180 PRINT BL$; CD$; CD$; EL$; CD$; TAB(5)"FIXING REFERENCE POINTERS..."
2190 FOR I=M+1 TO K1: IF INT(A6$(I)/10)>M THEN A6$(I)=A6$(I)+10: NEXT ELSE NEXT
2200 IF K2<0 THEN FOR I=1 TO K2: IF C5$(I)>M THEN C5$(I)=C5$(I)+1: NEXT ELSE NEXT
2210 IF K3<0 THEN FOR I=1 TO K3: IF E4$(I)>M THEN E4$(I)=E4$(I)+1: NEXT ELSE NEXT
2220 GOTO2010
2500 PRINT BL$; CD$; CD$; EL$; : GOSUB5000: PRINT CU$; CU$; CU$; : RETURN

```

```

2600 PRINTBL$:CD$:CD$:EL$:CU$:CU$: :RETURN
3000 GOSUB350
3010 PRINT:PRINT"ENTER START VALUE":GOSUB700:J=VAL(B$):IFB$="END"RETURN
3020 PRINT:PRINT" ENTER INCREMENT":GOSUB700:PRINT:K=VAL(B$)
3030 IFJ<10RK<10RJ+(K1-1)*K>32000THEN3040ELSE3050
3040 E=6:GOSUB5000:GOTO3010
3050 PRINTTAB(5)"RENUMBERING..."
3060 FORI=1TOK1:A1%(I)=J*SGN(A1%(I)):J=J+K
3070 NEXT:GOSUB320:RETURN
5000 IFE=99PRINT"* END OF RUN *":END
5010 PRINTTAB(5):ONEGOTO5011,5012,5013,5014,5015,5016,5017,5018,5019
5011 PRINT"* FILE NOT FOUND *":RETURN
5012 PRINT"* INPUT DATA EXCEEDS MEMORY ALLOCATION *":RETURN
5013 PRINT"* BAD LEVEL *":RETURN
5014 PRINT"* BAD PRIORITY *":RETURN
5015 PRINT"* FATAL ERROR - ABORTED *":RETURN
5016 PRINT"* BAD ACCOUNT NUMBER SEQUENCE *":RETURN
5017 PRINT"* NO ROOM TO INSERT *":RETURN
5018 PRINT"* BAD ACCOUNT NUMBER *":RETURN
9000 GOSUB350:PRINT:IO=1:IFDK=1THENDV=1:ONERRORGOTO9900
9010 IFDK=1THENPRINT"FILESPEC OF OUTPUT FILE: ":GOSUB750:PRINT:OPEN"O",DV,B$:ON
ERRORGOTO0
9020 IFDK=0THENDV=-1:INPUT"LOAD DATA TAPE(RECORD), THEN PRESS ENTER":A$
9030 PRINT#DV,INT(DT/100):PRINT"WRITING ACCOUNTS":PRINT#DV,K1
9040 IFK1<0THENFORI=1TOK1+1:PRINT#DV,A1%(I);X;A2$(I);X;A3$(I);X;A4$(I);X;A6%(I)
:NEXT
9050 PRINT"WRITING CHECKS":PRINT#DV,K2
9060 IFK2<0THENFORI=1TOK2:PRINT#DV,C1%(I);X;C2%(I);X;C3$(I);X;C4(I);X;C5%(I);X;
C6%(I):NEXT
9070 PRINT"WRITING CASH EXPENSES":PRINT#DV,K3
9080 IFK3<0THENFORI=1TOK3:PRINT#DV,E1%(I);X;E2$(I);X;E3(I);X;E4%(I):NEXT
9090 PRINT"WRITING DEPOSITS":PRINT#DV,K4
9100 IFK4<0THENFORI=1TOK4:PRINT#DV,D1%(I);X;D2$(I);X;D3(I):NEXT
9110 PRINT"WRITING SAVINGS DATA":PRINT#DV,K5
9120 IFK5<0THENFORI=1TOK5:PRINT#DV,S1%(I);X;S2$(I);X;S3(I):NEXT
9130 F=1:IFDK=1THENCLOSE:RETURNELSERETURN
9400 POKE11,I2:RETURN
9500 JJ=J:J=4:GOSUB350:J=JJ:PRINT:IO=0:I1=15423:I2=32:IFDK=1THENDV=1:ONERRORGOTO
9900
9510 IFDK=1THENPRINT"FILESPEC OF INPUT FILE: ":GOSUB750:PRINT:OPEN"I",DV,B$:ONE
RRORGOTO0
9520 IFDK=0THENDV=-1:INPUT"LOAD DATA TAPE(PLAY), THEN PRESS ENTER":A$
9530 INPUT#DV,D:PRINT"DATE OF DATA: ":GOSUB950:PRINT:PRINT"READING ACCOUNTS
9540 INPUT#DV,K1:IFK1+1>M1THEN9700
9550 IFK1<0THENFORI=1TOK1+1:INPUT#DV,A1%(I),A2$(I),A3$(I),A4$(I),A6%(I):GOSUB94
00:NEXT
9560 PRINT"READING CHECKS":INPUT#DV,K2:IFK2>M2THEN9700
9570 IFK2<0THENFORI=1TOK2:INPUT#DV,C1%(I),C2%(I),C3$(I),C4(I),C5%(I),C6%(I):GOS
UB9400:NEXT
9580 PRINT"READING CASH EXPENSES":INPUT#DV,K3:IFK3>M3THEN9700
9590 IFK3<0THENFORI=1TOK3:INPUT#DV,E1%(I),E2$(I),E3(I),E4%(I):GOSUB9400:NEXT
9600 PRINT"READING DEPOSITS":INPUT#DV,K4:IFK4>M4THEN9700
9610 IFK4<0THENFORI=1TOK4:INPUT#DV,D1%(I),D2$(I),D3(I):GOSUB9400:NEXT

```



```
=====
9620 PRINT"READING SAVINGS DATA":INPUT#DV,K5:IFK5>M5THEN9700
9630 IFK5<>0THENFORI=1TOK5:INPUT#DV,S1%(I),S2%(I),S3(I):GOSUB9400:NEXT
9640 F=1:IFDK=1THENCLOSE:RETURNELSERETURN
9700 E=2:GOSUB5000:STOP
9900 I=ERR/2+1:IFI=54THENE=1:GOSUB5000:IFI0=1THENRESUME9010ELSERESUME9510
9910 PRINT:PRINT"UNEXPECTED ERROR #"I"IN LINE"ERL"-- RUN ABORTED":END
10000 DATA CHECKBOOK BALANCE,0,SAVINGS ACCOUNT,0,PETTY CASH,0,TOTAL INCOME,0,REG
    ULAR INCOME,1
10010 DATA WAGE/SALARY,2,BUSINESS INCOME,2,SOC. SEC. BENIFITS,2,PENSION,2
10020 DATA VARIABLE INCOME,1,BONUS,2,INTEREST,2,SAVINGS,3,BONDS,3,DIVIDENDS,2
10030 DATA OTHER INCOME,2,SET-ASIDE INCOME,0,FUTURE GOALS,1,EMERGENCIES,1
10040 DATA EDUCATION,1,RETIREMENT,1,VACATION,1,TOTAL EXPENSES,0,REGULAR EXPENSES
    ,1
10050 DATA RENT/MORTGAGE,2,HOUSE MAINTENANCE,2,UTILITIES,2,ELECTRICITY,3,NATURAL
    GAS,3
10060 DATA WATER,3,TELEPHONE,2,INSTALLMENT PAYMENTS,2,INSURANCE,2,FOOD & BEVERAG
    E,2
10070 DATA FURNISHINGS,2,CLOTHING,2,TRANSPORTATION,2,BUS,3,CAR,3,GAS,4,CAR MAINTE
    NANCE,4
10080 DATA MEDICAL,2,RECREATION,2,MISCELLANEOUS,2,SEASONAL EXPENSES,1,EDUCATION,
    2,CONTRIBUTIONS,2
10090 DATA TAXES,2,CHRISTMAS,2,BIRTHDAYS,2,TRAVEL,2,"",-1
```

```

10 / BUDGET MANAGEMENT - TRANSACTION 1.0
20 / COPYRIGHT (C) 1979 TANDY CORP.
100 CLEAR(3600):DEFINTI-N:DEFSTRX
110 M1=61:M2=50:M3=25:M4=20:M5=M4
120 X="":AD$=CHR$(25):EL$=CHR$(30):CU$=CHR$(27):BL$=CHR$(29)
130 DIMA1%(M1),A2$(M1),A3$(M1),A4$(M1),A6$(M1)
140 DIMC1%(M2),C2$(M2),C3$(M2),C4$(M2),C5$(M2),C6$(M2)
150 DIME1%(M3),E2$(M3),E3$(M3),E4$(M3)
160 DIMD1%(M4),D2$(M4),D3$(M4)
170 DIMS1%(M5),S2$(M5),S3$(M5)
190 ONERRORGOTO200:DK=1:I=INSTR("A","A"):GOTO210
200 DK=0:RESUME210
210 ONERRORGOTO0
220 CLS:PRINT:PRINTTAB(10)"BUDGET MANAGEMENT - TRANSACTION 1.0":PRINT
230 IFDT<10100PRINT"ENTER TODAY'S DATE (MM/DD/YY)":X%=1:GOSUB900:DT=VAL(DT$):X%
=0:GOTO220
240 PRINTTAB(15)"0) STOP.
    "TAB(15)"1) ENTER DEPOSITS
    "TAB(15)"2) ENTER CASH EXPENSES
    "TAB(15)"3) ENTER OUTSTANDING CHECKS
    "TAB(15)"4) CANCEL CHECKS
    "TAB(15)"5) ENTER INCOME AMOUNTS
250 PRINTTAB(15)"6) ENTER SAVINGS TRANSACTIONS
    "TAB(15)"7) TRANSFER FUNDS
    "TAB(15)"8) START NEW MONTH
    "TAB(15)"9) SAVE DATA
260 PRINT:PRINT"ENTER YOUR CHOICE":GOSUB700:PRINT:J=VAL(B$):IFB$=""THEN220
270 IFJ=0THENC$="":PRINT"HAVE YOU SAVED YOUR DATA (Y/N)":GOSUB300:IFYN=0THEN220
    ELSEE=99:GOSUB5500
275 IFJ<00RJ>9THEN220ELSEIFF=0THENGOSUB9500
280 ONJGOSUB4000,3000,1000,2000,6000,5000,7000,800,9000:GOTO220
300 YN=-1:GOSUB700:PRINT:IFB$=""THENB$=C$
305 C$=LEFT$(B$,1):IFC$="Y"THENYN=1:RETURNELSEIFC$="N"THENYN=0:RETURN
310 PRINT"WAKE UP! PLEASE ANSWER YES(Y), OR NO(N)":IFF1=1THENRETURN:ELSE300
320 FORDY=1TO1000:NEXT:RETURN
350 CLS:PRINTTAB(18):ONJGOTO354,353,351,352,356,355,357,358,359
351 PRINT"OUTSTANDING CHECKS":RETURN
352 PRINT"CANCELLED CHECKS":RETURN
353 PRINT"CASH EXPENSES":RETURN
354 PRINT"DEPOSITS":RETURN
355 PRINT"SAVINGS TRANSACTIONS":RETURN
356 PRINT"INCOME ACCOUNTS":RETURN
357 PRINT"TRANSFER FUNDS":RETURN
358 PRINT"NEW MONTH":RETURN
359 PRINT"DATA INPUT/OUTPUT":RETURN
360 IFJ=3THENPRINT" NO. ";
363 PRINTTAB(10)"DATE"TAB(22)"DESCRIPTION"TAB(42)"AMOUNT";
367 IFJ=20RJ=3THENPRINTTAB(52)"EXP. ACCT":RETURN:ELSEPRINT:RETURN
400 PRINTEL$:IFJ=3THENGOSUB700:V=VAL(B$):IFB$="END"ORV=0THENRETURN
405 X%=0:PRINTBL$:STRING$(8,AD$):EL$:GOSUB900:IFB$<" "THENIFB$="END"THENW=0:RET
    URNELSEIFDT$=" "THEN405ELSEW=VAL(DT$):GOTO410
407 W=INT(DT/100):D=W:PRINTBL$:STRING$(8,AD$):"? ":GOSUB950
410 PRINTBL$:STRING$(17,AD$):EL$:GOSUB700:X1$=B$

```

```

415 PRINTBL$: STRING$(41, AD$); EL$: GOSUB700: Y=VAL(B$)
420 IFJ=20RJ=3THENPRINTBL$: STRING$(53, AD$); EL$: GOSUB700: Z=VAL(B$)
425 PRINT: RETURN
450 LO=1: H%=K1: II=0
460 IFLO=H%ANDNN<>ABS(A1%(LO))THENRETURN
470 MD=INT((LO+H%)/2): IFNN=ABS(A1%(MD))THENII=MD: RETURN
480 IFNN>ABS(A1%(MD))THENLO=MD+1: GOTO460: ELSEH%=MD: GOTO460
700 PRINT"? ";
750 B$="": L=0: PRINTCHR$(14);
760 W$=INKEY$: IFW$=""THEN760
770 IFW$=CHR$(13)THENPRINTCHR$(15);: L=L+1: RETURN
780 IFW$=CHR$(8)THENIFB$<>""THENPRINTW$: B$=LEFT$(B$, LEN(B$)-1): GOTO760ELSE760
790 IFW$=""THENL=L+1: PRINT " ";: GOTO760
794 IFW$=""THENPRINTSTRING$(LEN(B$)+2*L, CHR$(8)); EL$: GOTO750
796 IFW$< " 0RW$>"Z"THEN760ELSEPRINTW$: B$=B$+W$: GOTO760
800 GOSUB350: PRINT: PRINT"HAVE YOU SAVED THIS MONTH'S DATA";: C$=""
810 GOSUB300: IFYN=0THENRETURN
820 PRINT:I=0: INPUT"TYPE 1 TO START A NEW MONTH"; I: IFI<>1RETURN
825 PRINT"CLEARING.... ": W#=0D0
830 K3=0: K4=0: K5=0: FORI=4TOK1: A3#(I)=W#: NEXT
840 K=0: FORI=1TOK2: IFC6%(I)=0THENK=K+1: C1%(K)=C1%(I): C2%(K)=C2%(I): C3%(K)=C3%(I)
: C4%(K)=C4%(I): C5%(K)=C5%(I): C6%(K)=0
850 NEXT: K2=K: RETURN
900 GOSUB700: IFLEN(B$)<>5+3*X%THEN930ELSED1$=LEFT$(B$, 2): D2$=MID$(B$, 4, 2): D3$=RI
GHT$(B$, 2): IFVAL(D1$)<10RVAL(D1$)>12THEN930
905 IFVAL(D2$)<10RVAL(D2$)>31THEN930ELSEIFX%=1THENIFVAL(D3$)<0THEN930
910 DT$=D1$+D2$: IFX%=1THENDT$=DT$+D3$: RETURNELSERETURN
930 DT$=" ": RETURN
950 PRINTUSING"##/##"; INT(D/100); D-100*INT(D/100);: RETURN
1000 GOSUB350
1010 PRINT: PRINTA2$(1);: PRINTUSING" $####, ###, ###. ##"; A3$(1)
1020 GOSUB360: GOSUB400: IFV=0THENRETURN
1025 IFK2<>0THENFORI=1TOK2: IFC1%(I)=VTHENE=2: GOSUB5500: GOTO1010ELSENEXT
1030 NN=Z: GOSUB450: IFII=0ORA1%(II)>0THENE=1: GOSUB5500: GOTO1010
1040 IFK2+1>M2THENPRINTTAB(5)*" CHECKS FULL *": GOTO1010
1050 K2=K2+1: C1%(K2)=V: C2%(K2)=W: C3%(K2)=X1$: C4%(K2)=Y: C5%(K2)=II: C6%(K2)=0
1060 A3$(1)=A3$(1)-CDBL(Y): K=II
1070 A3%(K)=A3%(K)+CDBL(Y): K=INT(A6%(K)/10): IFK<>0THEN1070ELSE1010
2000 GOSUB350
2010 PRINT: PRINT"ENTER CHECK NUMBER";: GOSUB700: PRINT: NN=VAL(B$): IFNN=0RETURN
2015 IFK2=0THENPRINTTAB(5)*" NO CHECKS *": GOTO2010
2020 FORI=1TOK2: IFC1%(I)=NNTHEN2030ELSENEXT: E=3: GOSUB5500: GOTO2010
2030 IFC6%(I)<>0THENPRINT"* CHECK ALREADY CANCELLED *": GOTO2010
2040 PRINTTAB(10); C2%(I); TAB(19); C3%(I); TAB(41);: PRINTUSING"$####, ###. ##"
: C4%(I); ABS(A1%(C5%(I)))
2050 C6%(I)=INT(DT/100): GOTO2010
3000 GOSUB350: GOSUB3100: PRINT"ENTER 'PETTY CASH' DEPOSIT (0 IF NONE)";: GOSUB700:
PRINT: IFB$="END"THENRETURNELSEA3$(3)=A3$(3)+CDBL(ABS(VAL(B$)))
3010 GOSUB3100: GOSUB360: GOSUB400: IFW=0RETURN
3020 NN=Z: GOSUB450: IFII=0ORA1%(II)>0THENE=1: GOSUB5500: GOTO3010
3030 IFK3+1>M3THENPRINTTAB(5)*" CASH EXPENSES FULL *": GOTO3010
3040 K3=K3+1: E1%(K3)=W: E2$(K3)=X1$: E3(K3)=Y: E4%(K3)=II: K=II: A3$(3)=A3$(3)-CDBL(Y)
)

```

```

3050 A3#(K)=A3#(K)+CDBL(Y):K=INT(A6%(K)/10):IFK<>0THEN3050ELSE3010
3100 PRINTCHR$(13);TAB(5);A2$(3);:PRINTUSING"  $####,###,###.##";A3$(3):RETURN
4000 GOSUB350
4010 PRINT:PRINTA2$(1);:PRINTUSING"  $####,###,###.##";A3$(1)
4020 GOSUB360:GOSUB400:IFW=0THENRETURN
4030 IFK4+1>M4THENPRINTTAB(5)"* DEPOSITS FULL *":GOTO4010
4040 K4=K4+1:D1%(K4)=W:D2$(K4)=X1$:D3(K4)=ABS(Y):A3$(1)=A3$(1)+CDBL(ABS(Y)):GOTO
4010
5000 GOSUB350
5010 PRINT:PRINTA2$(2);:PRINTUSING"  $####,###,###.##";A3$(2)
5020 GOSUB360:GOSUB400:IFW=0RETURN
5030 IFK5+1>M5THENPRINTTAB(5)"* SAVINGS DATA FULL *":GOTO5010
5040 K5=K5+1:S1%(K5)=W:S2$(K5)=X1$:S3(K5)=Y:A3$(2)=A3$(2)+CDBL(Y):GOTO5010
5500 IFE=99THENPRINT"* END OF RUN *":END
5510 PRINTTAB(5);:ONEGOTO5511,5512,5513,5514,5515,5516,5517,5518,5519
5511 PRINT"* BAD EXPENSE ACCT *":RETURN
5512 PRINT"* CHECK EXISTS *":RETURN
5513 PRINT"* CHECK NOT FOUND *":RETURN
5514 PRINT"* FILE NOT FOUND *":RETURN
5515 PRINT"* INPUT DATA EXCEEDS MEMORY ALLOCATION *":RETURN
6000 GOSUB350:PRINT:PRINTTAB(40)"AMOUNT":FORI=4TOK1:L=INT(A6%(I)-10*INT(A6%(I)/1
0))
6005 IFA2$(I)="TOTAL EXPENSES"THENPRINTTAB(5)"* DONE *":GOSUB320:RETURN
6010 PRINTUSING"##### %"+STRING$(30," ")+"%";ABS(A1%(I));STRING$(L*2," ")+A2$(I)
+" "+STRING$(32," ");
6020 IFA1%(I)>0THENPRINT:NEXT
6030 PRINTTAB(40);:GOSUB700:PRINT:IFB$="END"THENRETURNELSEV=ABS(VAL(B$)):IFV=0TH
ENNEXTELSEK=I
6040 A3$(K)=A3$(K)+CDBL(Y):K=INT(A6%(K)/10):IFK<>0THEN6040ELSENEXT
7000 GOSUB350
7010 PRINT:PRINT" TO ACCOUNT NUMBER":GOSUB700:V=VAL(B$):IFB$="END"RETURN
7020 NN=V:GOSUB450:IFII=0THENPRINT:E=1:GOSUB5500:GOTO7010ELSEGOSUB7100
7030 I=II:PRINT"FROM ACCOUNT NUMBER":GOSUB700:V=VAL(B$):IFB$="END"THENRETURNELS
EIFV=0THENPRINT:GOTO7050
7040 NN=V:GOSUB450:IFII=0THENPRINT:E=1:GOSUB5500:GOTO7010ELSEGOSUB7100
7050 PRINT" AMOUNT":GOSUB700:PRINT:IFB$="END"THENRETURNELSEW=CDBL(
VAL(B$)):IFW=0D0THEN7010ELSEK=I
7060 A3$(K)=A3$(K)+W#K=INT(A6%(K)/10):IFK<>0THEN7060ELSEIFV=0THEN7010ELSEK=II
7080 A3$(K)=A3$(K)-W#K=INT(A6%(K)/10):IFK<>0THEN7080ELSE7010
7100 PRINTBL$;STRING$(28,AD$);EL$;A2$(II);TAB(50);:PRINTUSING"$####,###.##";A3$(
II):RETURN
9000 GOSUB350:PRINT:I0=1:IFDK=1THENDV=1:ONERRORGOTO9900
9010 IFDK=1THENPRINT"FILESPEC OF OUTPUT FILE: ";:GOSUB750:PRINT:OPEN"O",DV,B$:ON
ERRORGOTO0
9020 IFDK=0THENDV=-1:INPUT"LOAD DATA TAPE(RECORD), THEN PRESS ENTER";A$
9030 PRINT#DV,INT(DT/100):PRINT"WRITING ACCOUNTS":PRINT#DV,K1
9040 IFK1<>0THENFORI=1TOK1+1:PRINT#DV,A1%(I);X;A2$(I);X;A3$(I);X;A4$(I);X;A6%(I)
:NEXT
9050 PRINT"WRITING CHECKS":PRINT#DV,K2
9060 IFK2<>0THENFORI=1TOK2:PRINT#DV,C1%(I);X;C2%(I);X;C3$(I);X;C4(I);X;C5%(I);X;
C6%(I):NEXT
9070 PRINT"WRITING CASH EXPENSES":PRINT#DV,K3
9080 IFK3<>0THENFORI=1TOK3:PRINT#DV,E1%(I);X;E2$(I);X;E3(I);X;E4%(I):NEXT

```

```
=====
9090 PRINT"WRITING DEPOSITS":PRINT#DV,K4
9100 IFK4<>0THENFORI=1TOK4:PRINT#DV,D1%(I);X;D2%(I);X;D3(I):NEXT
9110 PRINT"WRITING SAVINGS DATA":PRINT#DV,K5
9120 IFK5<>0THENFORI=1TOK5:PRINT#DV,S1%(I);X;S2%(I);X;S3(I):NEXT
9130 F=1:IFDK=1THENCLOSE:RETURNELSERETURN
9400 POKEI1,I2:RETURN
9500 JJ=J:J=9:GOSUB350:J=JJ:PRINT:I0=0:I1=15423:I2=32:IFDK=1THENDV=1:ONERRORGOTO
9900
9510 IFDK=1THENPRINT"FILESPEC OF INPUT FILE: ";GOSUB750:PRINT:OPEN"I",DV,B$:ONE
RRORGOTO0
9520 IFDK=0THENDV=-1:INPUT"LOAD DATA TAPE(PLAY), THEN PRESS ENTER";A$
9530 INPUT#DV,D:PRINT"DATE OF DATA: ";GOSUB950:PRINT:PRINT"READING ACCOUNTS
9540 INPUT#DV,K1:IFK1+1>M1THEN9700
9550 IFK1<>0THENFORI=1TOK1+1:INPUT#DV,A1%(I),A2%(I),A3%(I),A4%(I),A6%(I):GOSUB94
00:NEXT
9560 PRINT"READING CHECKS":INPUT#DV,K2:IFK2>M2THEN9700
9570 IFK2<>0THENFORI=1TOK2:INPUT#DV,C1%(I),C2%(I),C3%(I),C4(I),C5%(I),C6%(I):GOS
UB9400:NEXT
9580 PRINT"READING CASH EXPENSES":INPUT#DV,K3:IFK3>M3THEN9700
9590 IFK3<>0THENFORI=1TOK3:INPUT#DV,E1%(I),E2%(I),E3(I),E4%(I):GOSUB9400:NEXT
9600 PRINT"READING DEPOSITS":INPUT#DV,K4:IFK4>M4THEN9700
9610 IFK4<>0THENFORI=1TOK4:INPUT#DV,D1%(I),D2%(I),D3(I):GOSUB9400:NEXT
9620 PRINT"READING SAVINGS DATA":INPUT#DV,K5:IFK5>M5THEN9700
9630 IFK5<>0THENFORI=1TOK5:INPUT#DV,S1%(I),S2%(I),S3(I):GOSUB9400:NEXT
9640 F=1:IFDK=1THENCLOSE:RETURNELSERETURN
9700 E=5:GOSUB5500:STOP
9900 I=ERR/2+1:IFI=54THENE=4:GOSUB5500:IFI0=1THENRESUME9010ELSERESUME9510
9910 PRINT:PRINT"UNEXPECTED ERROR #"I"IN LINE"ERL"-- RUN ABORTED":END
```

```

10 / BUDGET MANAGEMENT - EDIT 1.0
20 / COPYRIGHT (C) 1979 TANDY CORP.
100 CLEAR(3600):DEFINTI-N:DEFSTRX
110 M1=61:M2=50:M3=25:M4=20:M5=M4
120 X="," :AD$=CHR$(25):EL$=CHR$(30):CU$=CHR$(27):BL$=CHR$(29)
130 DIMA1%(M1),A2$(M1),A3$(M1),A4$(M1),A6$(M1)
140 DIMC1%(M2),C2$(M2),C3$(M2),C4$(M2),C5$(M2),C6$(M2)
150 DIME1%(M3),E2$(M3),E3$(M3),E4$(M3)
160 DIMD1%(M4),D2$(M4),D3$(M4)
170 DIMS1%(M5),S2$(M5),S3$(M5)
190 ONERRORGOTO200:DK=1:I=INSTR("A","A"):GOTO210
200 DK=0:RESUME210
210 ONERRORGOTO0
220 CLS:PRINT:PRINTTAB(10)"BUDGET MANAGEMENT - EDIT 1.0":PRINT
230 IFDT<10100PRINT"ENTER TODAY'S DATE (MM/DD/YY)":X%=1:GOSUB900:DT=VAL(DT$):X%
  =0:GOTO220
240 PRINTTAB(15)"0) STOP
    "TAB(15)"1) EDIT ACCOUNTS
    "TAB(15)"2) EDIT OUTSTANDING CHECKS
    "TAB(15)"3) EDIT CANCELLED CHECKS
    "TAB(15)"4) EDIT CASH EXPENSES
    "TAB(15)"5) EDIT DEPOSITS
    "TAB(15)"6) EDIT SAVINGS DATA
    "TAB(15)"7) SAVE DATA
260 PRINT:PRINT"ENTER YOUR CHOICE":GOSUB700:PRINT:J=VAL(B$)
270 IFJ=0THENPRINT"HAVE YOU SAVED YOUR DATA (Y/N)":GOSUB300:IFYN=0THEN220ELSE=
  99:GOSUB5500
280 IFJ<10RJ>7THEN220ELSEIFF=0THENGOSUB9500
285 IFJ=7THENGOSUB9000ELSEGOSUB2000
290 GOTO220
300 YN=-1:GOSUB700:PRINT:IFB$=""THENB$=C$
305 C$=LEFT$(B$,1):IFC$="Y"THENYN=1:RETURNELSEIFC$="N"THENYN=0:RETURN
310 PRINT"WAKE UP! PLEASE ANSWER YES(Y), OR NO(N)":IFF1=1THENRETURN:ELSE300
320 FORDY=1TO1000:NEXT:RETURN
350 CLS:PRINTTAB(18):ONJGOTO351,352,353,354,355,356,357
351 PRINT"ACCOUNT":GOTO360
352 PRINT"OUTSTANDING CHECKS":GOTO360
353 PRINT"CANCELLED CHECKS":GOTO360
354 PRINT"CASH EXPENSES":GOTO360
355 PRINT"DEPOSITS":GOTO360
356 PRINT"SAVINGS":GOTO360
357 PRINT"DATA INPUT/OUTPUT":RETURN
360 PRINT" EDITING":RETURN
400 I=KK
410 A3$(I)=A3$(I)+A#:A4$(I)=A4$(I)+B#:I=INT(A6$(I)/10):IFI<>0THEN410ELSEReturn
450 LO=1:H%=K:II=0
460 IFLO=H%ANDNN<>ABS(A1%(LO))THENRETURN
470 MD=INT((LO+H%)/2):IFNN=ABS(A1%(MD))THENII=MD:RETURN
480 IFNN>ABS(A1%(MD))THENLO=MD+1:GOTO460:ELSEH%=MD:GOTO460
500 E=0:FORI=1TOK2:IFA=C1%(I)THENE=4:GOSUB1800:RETURN
505 NEXT:RETURN
560 D2=INT(D/100):D3=D-100*D2:RETURN
600 GOSUB650:E=0:IFB$=""THENE=1:RETURNELSEReturn

```



```

650 GOSUB800
700 PRINT"? ";
750 B$="":PRINTCHR$(14);
760 W$=INKEY$:IFW$=""THEN760
770 IFW$=CHR$(13)THENPRINTCHR$(15);:RETURN
780 IFW$=CHR$(8)THENIFB$<>" "THENPRINTW$;B$=LEFT$(B$,LEN(B$)-1):GOTO760ELSE760
794 IFW$=""THENPRINTSTRING$(LEN(B$),CHR$(8));EL$:GOTO750
796 IFW$<" "ORW$>"Z"THEN760ELSEPRINTW$;B$=B$+W$:GOTO760
800 PRINTBL$;STRING$(P,AD$);EL$;:RETURN
900 GOSUB700:IFLEN(B$)>5+3*X%THEN930ELSE1$=LEFT$(B$,2):D2$=MID$(B$,4,2):D3$=RI
    GHT$(B$,2):IFVAL(D1$)<10RVAL(D1$)>12THEN930
905 IFVAL(D2$)<10RVAL(D2$)>31THEN930ELSEIFX%=1THENIFVAL(D3$)<0THEN930
910 DT$=D1$+D2$:IFX%=1THENDT$=DT$+D3$:RETURNELSERETURN
930 DT$=" ":RETURN
950 PRINTUSING"##/##";INT(D/100);D-100*INT(D/100);:RETURN
1000 PRINT">";CHR$(14);
1002 IFDK=1CMD"T"
1005 A$=INKEY$:IFA$=""THEN1005ELSEIFDK=1CMD"R"
1010 IFA$="X"THENEOD=1:ONERRORGOTO0:RETURN
1020 IFA$=CHR$(10)THENIC=1:PRINTCHR$(92):RETURN
1030 IFA$=CHR$(91)THENIC=-1:PRINTA$:RETURN
1040 IFA$<"E"THEN1002
1050 ONERRORGOTO1900:PRINTA$:C$=" "+STRING$(23," ")+%"
1060 ONJGOTO1100,1200,1200,1400,1500,1600
1100 P=1:GOSUB800:PRINTUSING"#####";ABS(A1%(KK));
1110 P=7:GOSUB600:IFE=0THENA2$(KK)=B$
1112 B$=STRING$(2*L," ")+A2$(KK)
1115 GOSUB800:PRINTUSINGC$;B$;
1120 P=33:GOSUB600:IFE<0THEN1130
1125 Z#=ABS(CDBL(VAL(B$))):A#=Z#-A3$(KK):B#=0D0:GOSUB400:A3$(KK)=Z#
1130 GOSUB800:PRINTUSING"#####.##";A3$(KK);
1135 P=43:GOSUB600:IFE<0THEN1145
1140 Z#=ABS(CDBL(VAL(B$))):A#=0D0:B#=Z#-A4$(KK):GOSUB400:A4$(KK)=Z#
1145 GOSUB800:PRINTUSING"#####.##";A4$(KK):IC=1:RETURN
1200 P=0:GOSUB600:IFE=0THENA=ABS(VAL(B$)):GOSUB500:IFE<0THEN1200ELSEC1%(KK)=A
1205 GOSUB800:PRINTUSING"#####";C1%(KK);:IFJ=2THENP=10ELSEP=8
1210 D=C2%(KK):GOSUB1700:IFB$=""THEN1215ELSEIFDT$=" "THEN1210ELSEC2%(KK)=D
1215 GOSUB1750:IFJ=2THEN1230
1220 P=14:D=C6%(KK):GOSUB1700:IFB$=""THEN1225ELSEIFDT$=" "THEN1220ELSEC6%(KK)=D
1225 GOSUB1750
1230 P=21:GOSUB600:IFE=0THENC3$(KK)=B$
1235 GOSUB800:PRINTUSINGC$;C3$(KK);
1240 P=42:GOSUB600:Z=C4$(KK):IFE=0THENZ=VAL(B$)
1245 GOSUB800:PRINTUSING"#####.##";Z;
1250 P=54:GOSUB600:NN=ABS(A1%(C5%(KK))):IFE=0THENNN=ABS(VAL(B$))
1255 K=K1:GOSUB450:IFII=0ORA1%(II)>0THENE=5:GOSUB1800:GOTO1250
1260 GOSUB800:PRINTUSING"#####";NN
1270 B#=0D0:A#=-CDBL(C4$(KK)):K=KK:KK=C5%(K):GOSUB400
1275 A#=CDBL(Z):KK=II:GOSUB400:KK=K
1280 A3$(1)=A3$(1)+CDBL(C4$(KK))-CDBL(Z):C4$(KK)=Z:C5%(KK)=II:IC=1:RETURN
1400 P=10:D=E1%(KK):GOSUB1700:IFB$=""THEN1405ELSEIFDT$=" "THEN1400ELSEE1%(KK)=D
1405 GOSUB1750
1410 P=21:GOSUB600:IFE=0THENE2$(KK)=B$

```

```

1415 GOSUB800:PRINTUSINGC$;E2$(KK);
1420 P=42:GOSUB600:Z=E3(KK):IFE=0THENZ=VAL(B$)
1425 GOSUB800:PRINTUSING"#####.##";Z;
1430 P=54:GOSUB600:NN=ABS(A1%(E4%(KK))):IFE=0THENNN=ABS(VAL(B$))
1435 K=K1:GOSUB450:IFII=0ORA1%(II)>0THENE=5:GOSUB1800:GOTO1430
1440 GOSUB800:PRINTUSING"#####";NN
1445 B#=0D0:A#=-CDBL(E3(KK)):K=KK:KK=E4%(K):GOSUB400:A3#(3)=A3#(3)-A#
1450 A#=CDBL(Z):KK=II:GOSUB400:KK=K:A3#(3)=A3#(3)-A#
1455 E3(KK)=Z:E4%(KK)=II:IC=1:RETURN
1500 P=10:D=D1%(KK):GOSUB1700:IFB$=""THEN1505ELSEIFDT$=""THEN1500ELSE1%(KK)=D
1505 GOSUB1750
1510 P=21:GOSUB600:IFE=0THEND2$(KK)=B$
1515 GOSUB800:PRINTUSINGC$;D2$(KK);
1520 P=42:GOSUB600:Z=D3(KK):IFE=0THENZ=ABS(VAL(B$))
1525 GOSUB800:PRINTUSING"#####.##";Z;
1530 A3#(1)=A3#(1)-CDBL(D3(KK))+CDBL(Z):D3(KK)=Z:IC=1:RETURN
1600 P=10:D=S1%(KK):GOSUB1700:IFB$=""THEN1605ELSEIFDT$=""THEN1600ELSE1%(KK)=D
1605 GOSUB1750
1610 P=21:GOSUB600:IFE=0THENS2$(KK)=B$
1615 GOSUB800:PRINTUSINGC$;S2$(KK);
1620 P=42:GOSUB600:Z=S3(KK):IFE=0THENZ=VAL(B$)
1625 GOSUB800:PRINTUSING"#####.##-";Z;
1630 A3#(2)=A3#(2)-CDBL(S3(KK))+CDBL(Z):S3(KK)=Z:IC=1:RETURN
1700 GOSUB800:GOSUB900:IFB$=""ORDT$=""THENRETURNS=ABS(VAL(DT$)):RETURN
1750 GOSUB800:GOSUB950:RETURN
1800 PRINTCHR$(13);BL$;:GOSUB5500:PRINTCU$;CU$;:RETURN
1900 E=6:GOSUB1800:RESUMENEXT
2000 GOSUB350:GOSUB3000:PRINTT$
2360 GOSUB4500:IFE0D=1PRINT:E=3:GOSUB5500:GOSUB320:RETURN
2380 GOSUB6000:GOSUB1000
2390 IFE0D=1RETURNELSE2360
3000 E0D=0:IC=1:KK=0:IFJ<>1THEN3060
3040 T$=" NO. NAME CURRENT BUDGET
3050 F$=" ##### % % #####.## #####.##":RETURN
3060 T$="DATE DESCRIPTION AMOUNT":F$="##/## %
% #####.##-
3070 IFJ>1ANDJ<4THENT$=" NO. "+T$:F$="##### "+F$:ELSET$=STRING$(10," ")
+T$:F$=STRING$(10," ")+F$
3080 IFJ>1ANDJ<5THENT$=T$+" EXP. ACCT":F$=F$+" #####"
3090 IFJ=3THENT$=" NO. WRTN CAN DESCRIPTION AMOUNT EXP. ACCT
":F$="##### ##/## ##/## % %#####.## #####":RETURNE
L$RETURN
4500 K=KK+IC:ONJGOTO4510,4520,4530,4540,4550,4560
4510 IFK1=0THEN4600ELSEIFK>K1THENRETURN
4515 IFK<10RKK=0THENKK=1:RETURNELSEKK=K:RETURN
4520 IFK2=0THEN4600ELSEIFK>K2THENRETURN
4522 IFK<1THENK=1
4524 IFIC=1THENFORI=KTOK2:IFC6%(I)=0THENKK=I:RETURNELSENEXT:GOTO4526
4525 IFIC=-1THENFORI=KT01STEP-1:IFC6%(I)=0THENKK=I:RETURNELSENEXT
4526 IFKK=0THEN4600ELSERETURN
4530 IFK2=0THEN4600ELSEIFK>K2THENRETURN
4532 IFK<1THENK=1
4534 IFIC=1THENFORI=KTOK2:IFC6%(I)<>0THENKK=I:RETURNELSENEXT:GOTO4536

```



```

4535 IFIC=-1THENFORI=KT01STEP-1:FC6%(I)<>0THENKK=I:RETURNELSENEXT
4536 IFKK=0THEN4600ELSERETURN
4540 IFK3=0THEN4600ELSEIFK>K3THENRETURNELSE4515
4550 IFK4=0THEN4600ELSEIFK>K4THENRETURNELSE4515
4560 IFK5=0THEN4600ELSEIFK>K5THENRETURNELSE4515
4600 E00=1:RETURN
5500 IFE=99PRINT"* END OF RUN *":END
5510 PRINTTAB(5);:ONEGOTO5511,5512,5513,5514,5515,5516,5517
5511 PRINT"* FILE NOT FOUND *":RETURN
5512 PRINT"* INPUT DATA EXCEEDS MEMORY ALLOCATION *":RETURN
5513 PRINT"* NO DATA *":RETURN
5514 PRINT"* CHECK EXISTS *":RETURN
5515 PRINT"* BAD EXPENSE ACCOUNT *":RETURN
5516 PRINT"* VALUE ERROR *":RETURN
6000 ONJGOTO6010,6001,6001,6003,6004,6005
6001 D=C2%(KK)
6002 IFJ=3THENGOSUB560:DY=D2:YN=D3:D=C6%(KK):GOTO6006ELSE6006
6003 D=E1%(KK):GOTO6006
6004 D=D1%(KK):GOTO6006
6005 D=S1%(KK)
6006 GOSUB560:ONJGOTO6010,6020,6030,6040,6050,6060
6010 L=A6%(KK)-10*INT(A6%(KK)/10):B$=STRING$(2*L,"")+A2$(KK)
6015 PRINTUSINGF$;ABS(A1%(KK));B$;A3$(KK);A4$(KK);:RETURN
6020 PRINTUSINGF$;C1%(KK);D2;D3;C3$(KK);C4(KK);ABS(A1%(C5%(KK)));:RETURN
6030 PRINTUSINGF$;C1%(KK);DY;YN;D2;D3;C3$(KK);C4(KK);ABS(A1%(C5%(KK)));:RETURN
6040 PRINTUSINGF$;D2;D3;E2$(KK);E3(KK);ABS(A1%(E4%(KK)));:RETURN
6050 PRINTUSINGF$;D2;D3;D2$(KK);D3(KK);:RETURN
6060 PRINTUSINGF$;D2;D3;S2$(KK);S3(KK);:RETURN
9000 GOSUB350:PRINT:I0=1:IFDK=1THENDV=1:ONERRORGOTO9900
9010 IFDK=1THENPRINT"FILESPEC OF OUTPUT FILE: ";:GOSUB750:PRINT:OPEN"O",DV,B$:ON
  ERRORGOTO0
9020 IFDK=0THENDV=-1:INPUT"LOAD DATA TAPE(RECORD), THEN PRESS ENTER";A$
9030 PRINT#DV,INT(DT/100):PRINT"WRITING ACCOUNTS":PRINT#DV,K1
9040 IFK1<>0THENFORI=1TOK1+1:PRINT#DV,A1%(I);X;A2$(I);X;A3$(I);X;A4$(I);X;A6%(I)
  :NEXT
9050 PRINT"WRITING CHECKS":PRINT#DV,K2
9060 IFK2<>0THENFORI=1TOK2:PRINT#DV,C1%(I);X;C2%(I);X;C3$(I);X;C4(I);X;C5%(I);X;
  C6%(I):NEXT
9070 PRINT"WRITING CASH EXPENSES":PRINT#DV,K3
9080 IFK3<>0THENFORI=1TOK3:PRINT#DV,E1%(I);X;E2$(I);X;E3(I);X;E4%(I):NEXT
9090 PRINT"WRITING DEPOSITS":PRINT#DV,K4
9100 IFK4<>0THENFORI=1TOK4:PRINT#DV,D1%(I);X;D2$(I);X;D3(I):NEXT
9110 PRINT"WRITING SAVINGS DATA":PRINT#DV,K5
9120 IFK5<>0THENFORI=1TOK5:PRINT#DV,S1%(I);X;S2$(I);X;S3(I):NEXT
9130 F=1:IFDK=1THENCLOSE:RETURNELSERETURN
9400 POKE11,I2:RETURN
9500 JJ=J:J=7:GOSUB350:J=JJ:PRINT:I0=0:I1=15423:I2=32:IFDK=1THENDV=1:ONERRORGOTO
  9900
9510 IFDK=1THENPRINT"FILESPEC OF INPUT FILE: ";:GOSUB750:PRINT:OPEN"I",DV,B$:ON
  ERRORGOTO0
9520 IFDK=0THENDV=-1:INPUT"LOAD DATA TAPE(PLAY), THEN PRESS ENTER";A$
9530 INPUT#DV,D:PRINT"DATE OF DATA: ";:GOSUB950:PRINT:PRINT"READING ACCOUNTS
9540 INPUT#DV,K1:IFK1+1>M1THEN9700

```

```
=====
9550 IFK1<>0THENFORI=1TOK1+1:INPUT#DV,A1%(I),A2$(I),A3$(I),A4$(I),A6%(I):GOSUB94
00:NEXT
9560 PRINT"READING CHECKS":INPUT#DV,K2:IFK2>M2THEN9700
9570 IFK2<>0THENFORI=1TOK2:INPUT#DV,C1%(I),C2%(I),C3$(I),C4$(I),C5%(I),C6%(I):GOS
UB9400:NEXT
9580 PRINT"READING CASH EXPENSES":INPUT#DV,K3:IFK3>M3THEN9700
9590 IFK3<>0THENFORI=1TOK3:INPUT#DV,E1%(I),E2$(I),E3$(I),E4%(I):GOSUB9400:NEXT
9600 PRINT"READING DEPOSITS":INPUT#DV,K4:IFK4>M4THEN9700
9610 IFK4<>0THENFORI=1TOK4:INPUT#DV,D1%(I),D2$(I),D3$(I):GOSUB9400:NEXT
9620 PRINT"READING SAVINGS DATA":INPUT#DV,K5:IFK5>M5THEN9700
9630 IFK5<>0THENFORI=1TOK5:INPUT#DV,S1%(I),S2$(I),S3$(I):GOSUB9400:NEXT
9640 F=1:IFDK=1THENCLOSE:RETURNELSERETURN
9700 E=2:GOSUB5500:STOP
9900 I=ERR/2+1:IFI=54THENE=1:GOSUB5500:IFI0=1THENRESUME9010ELSERESUME9510
9910 PRINT:PRINT"UNEXPECTED ERROR #"I"IN LINE"ERL"-- RUN ABORTED":END
```

```

10 / BUDGET MANAGEMENT - REPORT 1.0
20 / COPYRIGHT (C) 1979 TANDY CORP.
100 CLEAR(3600):DEFINTI-N
110 M1=61:M2=50:M3=25:M4=20:M5=M4
120 AD$=CHR$(25):EL$=CHR$(30):CU$=CHR$(27):BL$=CHR$(29)
130 DIMA1%(M1),A2$(M1),A3$(M1),A4$(M1),A6$(M1)
140 DIMC1%(M2),C2$(M2),C3$(M2),C4$(M2),C5$(M2),C6$(M2)
150 DIME1%(M3),E2$(M3),E3$(M3),E4$(M3)
160 DIMD1%(M4),D2$(M4),D3$(M4)
170 DIMS1%(M5),S2$(M5),S3$(M5)
190 ONERRORGOTO200:DK=1:I=INSTR("A","A"):GOTO210
200 DK=0:RESUME210
210 ONERRORGOTO0
220 CLS:PRINT:PRINTTAB(10)"BUDGET MANAGEMENT - REPORT 1.0":PRINT
230 IFDT<10100PRINT"ENTER TODAY'S DATE (MM/DD/YY)";:X%=1:GOSUB900:DT=VAL(DT$):X%
=0:GOTO220
240 PRINTTAB(15)"0) STOP
    "TAB(15)"1) READ-IN MORE DATA
    "TAB(15)"2) REPORT
260 PRINT:J=3:INPUT"ENTER YOUR CHOICE";J
270 IFJ=0THENE=99:GOSUB5000
275 IF(J-1)*(J-2)<0THEN220ELSEIFF=0ANDJ=2GOSUB9500
280 ONJGOSUB9500,2000:GOTO220
290 C$=LEFT$(C$+" "+STRING$(30," "),30):RETURN
300 YN=-1:INPUTB$:IFB$=""THENB$=C$
305 C$=LEFT$(B$,1):IFC$="Y"THENYN=1:RETURNELSEIFC$="N"THENYN=0:RETURN
310 PRINT"WAKE UP! PLEASE ANSWER YES(Y), OR NO(N)";:IFF1=1THENRETURN:ELSE300
320 FORDY=1TO1000:NEXT:RETURN
400 CLS:PRINTTAB(18);:ONJGOTO401,402
401 PRINT"DATA INPUT":RETURN
402 PRINT"BUDGET REPORTER":RETURN
450 L0=1:H%=K:II=0
460 IFLO=H%AND NN<ABS(A1%(L0))RETURN
470 MD=INT((LO+H%)/2):IFNN=ABS(A1%(MD))II=MD:RETURN
480 IFNN>ABS(A1%(MD))THENL0=MD+1:GOTO460:ELSEH%=MD-1:GOTO460
500 D1=INT(D/1E4)
510 D2=INT((D-D1*1E4)/100):D3=D-1E4*D1-100*D2:RETURN
550 D1=0:GOTO510
900 INPUTB$
902 IFLEN(B$)<5+3*X%THEN930ELSED1$=LEFT$(B$,2):D2$=MID$(B$,4,2):D3$=RIGHT$(B$,2)
:IFVAL(D1$)<10RVAL(D1$)>12THEN930
905 IFVAL(D2$)<10RVAL(D2$)>31THEN930ELSEIFX%=1THENIFVAL(D3$)<0THEN930
910 DT$=D1$+D2$:IFX%=1THENDT$=DT$+D3$:RETURNELSERETURN
930 DT$="":RETURN
950 PRINTUSING"###/###":INT(D/100);D-100*INT(D/100);:RETURN
960 PRINTUSING"###/###/###":INT(DT/1E4);INT((DT-1E4*INT(DT/1E4))/100);DT-100*INT(DT
/100);:RETURN
1000 A$=INKEY$:IFA$<"@"RETURN
1010 AB=1:PRINT:PRINTTAB(5)"* ABORTED *":PRINT:IFIO=2LPRINT"*** ABORTED ***"
1020 IFIO=1THENGOSUB1500:RETURNELSERETURN
1500 IFDK=1CMD"T"
1510 PRINT@980,"PRESS ENTER TO CONTINUE";:INPUTA$
1520 IFDK=1THENCMD"R":RETURNELSERETURN

```

```

2000 GOSUB400:PRINT:PRINTTAB(5)"REPORT WILL GO TO
      "TAB(15)"0) RETURN TO MENU
      "TAB(15)"1) VIDEO DISPLAY
      "TAB(15)"2) LINE PRINTER
2020 PRINT:PRINT"ENTER YOUR CHOICE";:INPUTB$:IO=VAL(B$):IFI0=0ORB$="END"THENRETU
      RN
2025 IF(I0-1)*(I0-2)<>0THEN2000
2030 IFI0=2PRINT:INPUT"READY LINE PRINTER, THEN PRESS ENTER";A$
2040 GOSUB400:PRINT:PRINTTAB(5)"TYPE OF REPORT IS
      "TAB(15)"0) RETURN TO MENU
      "TAB(15)"1) ACCOUNTS
      "TAB(15)"2) OUTSTANDING CHECKS
      "TAB(15)"3) CANCELLED CHECKS
2050 PP=0:PRINTTAB(15)"4) CASH EXPENSES
      "TAB(15)"5) DEPOSITS
      "TAB(15)"6) SAVINGS
2080 PRINT:PRINT"ENTER YOUR CHOICE";:INPUTB$:TY=VAL(B$):IFTY=0ORB$="END"THENRETU
      RN
2085 IFTY<10RTY>6THEN2040
2090 GOSUB400:PRINT:PRINTTAB(5)"SEARCHING FOR
2100 PRINTTAB(15)"0) RETURN TO MENU
      "TAB(15)"1) CANCELLED DATE
      "TAB(15)"2) CHECK NUMBER
      "TAB(15)"3) EXPENSES ACCOUNT
      "TAB(15)"4) DATE ENTERED
2110 PRINTTAB(15)"5) DESCRIPTION
      "TAB(15)"6) AMOUNT
      "TAB(15)"7) ALL":LK=0:PRINT
2130 PRINT"ENTER YOUR CHOICE";:INPUTB$:LK=VAL(B$):IFB$="END"ORLK=0THENRETURN
2135 IFLK<10RLK>7THEN2090ELSEIFTY=3THEN2190
2140 IFTY=1AND(LK=30RLK>4)THEN2190
2150 IFTY=2ANDLK>1THEN2190
2160 IFTY=4ANDLK>2THEN2190
2170 IF(TY=50RTY=6)ANDLK>3THEN2190
2180 PRINTTAB(5)"* BAD REQUEST *":GOSUB320:GOTO2040
2190 PRINT:IFLK=5PRINT"ENTER DESCRIPTION";TAB(32);:INPUTR$:GOTO2300
2200 ONLKGOTO2201,2202,2203,2201,2300,2206,2300
2201 PRINT"<LOW DATE>, <HIGH DATE>";:GOTO2210
2202 PRINT"<LOW CHECK#>, <HIGH CHECK#>";:GOTO2210
2203 PRINT"<LOW ACCOUNT#>, <HIGH ACCOUNT#>";:GOTO2210
2206 PRINT"<LOW AMOUNT>, <HIGH AMOUNT>";
2210 PRINTTAB(32);:IFLK<>1ANDLK<>4THEN2250
2220 INPUTB$,C$:GOSUB902:IFDT$=" "THEN2270ELSER1=VAL(DT$)
2230 B$=C$:GOSUB902:IFDT$=" "THEN2270ELSER2=VAL(DT$):GOTO2260
2250 INPUTR1,R2
2260 IFR1<=R2THEN2300
2270 PRINTTAB(5)"* BAD RANGE *":GOTO2190
2300 IFI0=1THEN2330
2305 IFPEEK(14312)<128THEN2330
2320 PRINT:PRINT"LINE PRINTER NOT READY!!":INPUT"TYPE 1 WHEN READY, OR 0 TO RETU
      RN TO MENU";I:IFI=0THENRETURNELSE2305
2330 GOSUB3000

```

```

2340 GOSUB3500:GOSUB1000:IFAB=1THEN2040
2360 GOSUB4500:GOSUB1000:IFAB=1THEN2040
2365 IFEOD=1ANDPP=0THENH$="      * NO ENTRIES *":IFIO=1THENPRINTH$:ELSELPRINTH$
2370 IFEOD=1THEN2410
2380 GOSUB6000:PP=1:GOSUB1000:IFAB=1THEN2040
2390 IFIO=1THENCNT=CN+1:IFCNT>14THENGOSUB1500:GOTO2340:ELSE2360
2400 IFPEEK(16425)>62THENLPRINTCHR$(12):GOTO2340:ELSE2360
2410 IFIO=1GOSUB1500:GOTO2040
2420 LPRINTCHR$(12):GOTO2040
3000 POKE16424,67:POKE16425,1:PG=0:CN=0:EOD=0:KK=0:AB=0
3010 ONTYGOTO3011,3012,3013,3014,3015,3016
3011 H$="ACCOUNTS":GOTO3020
3012 H$="OUTSTANDING CHECKS":GOTO3020
3013 H$="CANCELLED CHECKS":GOTO3020
3014 H$="CASH EXPENSES":GOTO3020
3015 H$="DEPOSITS":GOTO3020
3016 H$="SAVINGS"
3020 IFIO=2THEN3100
3030 IFTY<>1THEN3060
3040 T$="  NO.      NAME      CURRENT  BUDGET  REMAINING
3050 F$="  ##### %      %  #####. ##  #####. ##  #####. ##-":RETURN
3060 T$="DATE      DESCRIPTION      AMOUNT":F$="##/##  %
      %  #####. ##-
3070 IFTY>1ANDTY<4T$="  NO.      "+T$:F$="#####  "+F$:ELSET$=STRING$(10," ")+T
      $:F$=STRING$(10," ")+F$
3080 IFTY>1ANDTY<5T$=T$+"  EXP. ACCT":F$=F$+"  #####"
3090 IFTY=3THENT$="  NO.  WRTN  CAN  DESCRIPTION      AMOUNT  EXP. AC
      CT":F$="#####  ##/##  ##/##  %      %  #####. ##  #####. ##-":RETU
      RNELSERETURN
3100 IFTY<>1THEN3130
3110 T$="  ACCOUNT      NAME      CURRENT  BUDGET
      PERCENT  REMAINING
3120 F$="  ##### %      %  ###. ###. ##  ###. ###. ##
      #####. ##  ###. ###. ##-":RETURN
3130 T$="DATE      DESCRIPTION      AMOUNT
3135 F$="##/##  %      %  ###. ###. ##-
3140 IFTY>1ANDTY<4THENT$="  NUMBER      "+T$:F$="#####  "+F$:ELSET$=STR
      ING$(15," ")+T$:F$=STRING$(15," ")+F$
3150 IFTY>1ANDTY<5T$=T$+"  EXPENSE":F$=F$+"  #####
3160 IFTY=3THENT$=T$+"  CANCELLED":F$=F$+"  ##/##":RETURNELSERETURN
3500 PG=PG+1:IFIO=2THEN3530
3510 CLS:D=DT:GOSUB500:PRINTUSING"BUDGET MANAGEMENT: %      % PAGE
      ##  DATE ##/##/##":H$:PG:D1:D2:D3
3520 GOSUB1000:IFAB=1THENRETURNELSEPRINTT$:CN=2:RETURN
3530 GOSUB3900:LPRINTTAB(40);"BUDGET MANAGEMENT REPORT"+STRING$(25," ");:LPRINTU
      SING"PAGE ##":PG
3540 LPRINTTAB(40)"**  H$  REPORT **":LPRINTTAB(45)"DATE: ";:D=DT:GOSUB500:LPRIN
      TUSING"##/##/##":D1:D2:D3
3550 GOSUB1000:IFAB=1THENRETURNELSEGOSUB3910:LPRINTT$:FORI=1TOLEN(T$):IFMID$(T$,
      I,1)=" "THENLPRINT" ";:ELSELPRINT"=";
3560 NEXT:LPRINT" ":RETURN
3900 LPRINTSTRING$(110,"-")
3910 FORDY=1TO2:LPRINT" ":NEXT:RETURN

```

```

4500 EOD=0:KK=KK+1:ONTYGOTO4540,4600,4680,4690,4740,4780
4510 EOD=1:RETURN
4540 C$=A2$(KK):IFKK>K1THEN4510ELSEIFLK=7THEN4580ELSEONLKGOTO4510,4510,4550,4510
,4560,4570
4550 K=ABS(A1$(KK)):IFK<INT(R1)ORR<INT(R2)THEN4500ELSE4580
4560 IFR$<>C$THEN4500ELSE4580
4570 R=CSNG(A3$(KK)):IFR<R1ORR>R2THEN4500ELSE4580
4580 IFA4$(KK)=0D0THENP=0:R#=0D0:ELSER#=(A4$(KK)-A3$(KK)):P=A3$(KK)/A4$(KK)*100
4590 L=INT(A6$(KK)-10*INT(A6$(KK)/10)):C$=STRING$(2*L,"")+A2$(KK)
4595 IFL=0ANDI0=2THENLPRINT" ":RETURNELSERETURN
4600 IFKK>K2THEN4510ELSEIFC6$(KK)<>0THEN4500
4610 C$=C3$(KK):IFLK=7THENRETURNELSEONLKGOTO4670,4620,4660,4630,4910,4650
4620 R=C1$(KK):GOTO4900
4630 R=C2$(KK):GOTO4900
4650 R=C4$(KK):GOTO4900
4660 R=ABS(A1$(C5$(KK))):GOTO4900
4670 R=C6$(KK):GOTO4900
4680 IFKK>K2THEN4510ELSEIFC6$(KK)=0THEN4500ELSE4610
4690 C$=E2$(KK):IFKK>K3THEN4510ELSEIFLK=7THENRETURNELSEONLKGOTO4510,4510,4730,47
00,4910,4720
4700 R=E1$(KK):GOTO4900
4720 R=E3$(KK):GOTO4900
4730 R=ABS(A1$(E4$(KK))):GOTO4900
4740 C$=D2$(KK):IFKK>K4THEN4510ELSEIFLK=7THENRETURNELSEONLKGOTO4510,4510,4510,47
50,4910,4770
4750 R=D1$(KK):GOTO4900
4770 R=D3$(KK):GOTO4900
4780 C$=S2$(KK):IFKK>K5THEN4510ELSEIFLK=7THENRETURNELSEONLKGOTO4500,4500,4500,47
90,4910,4810
4790 R=S1$(KK):GOTO4900
4810 R=S3$(KK):GOTO4900
4900 IFR<R1ORR>R2THEN4500ELSERETURN
4910 IFR$<>C$THEN4500ELSERETURN
5000 IFE=99PRINT"* END OF RUN *":END
5010 PRINTTAB(5):ONEGOTO5011,5012,5013,5014,5015,5016,5017
5011 PRINT"* FILE NOT FOUND *":RETURN
5012 PRINT"* INPUT DATA EXCEEDS MEMORY ALLOCATION *":RETURN
6000 GOSUB290:ONTYGOTO6006,6001,6001,6003,6004,6005
6001 D=C2$(KK)
6002 IFTY=3THENGOSUB550:DY=D2:YN=D3:D=C6$(KK):GOTO6006ELSE6006
6003 D=E1$(KK):GOTO6006
6004 D=D1$(KK):GOTO6006
6005 D=S1$(KK)
6006 GOSUB550:IFI0=2THEN6100ELSEONTYGOTO6010,6020,6030,6040,6050,6060
6010 PRINTUSINGF$;ABS(A1$(KK));C$;A3$(KK);A4$(KK);R#:RETURN
6020 PRINTUSINGF$;C1$(KK);D2;D3;C$;C4$(KK);ABS(A1$(C5$(KK))):RETURN
6030 PRINTUSINGF$;C1$(KK);DY;YN;D2;D3;C$;C4$(KK);ABS(A1$(C5$(KK))):RETURN
6040 PRINTUSINGF$;D2;D3;C$;E3$(KK);ABS(A1$(E4$(KK))):RETURN
6050 PRINTUSINGF$;D2;D3;C$;D3$(KK):RETURN
6060 PRINTUSINGF$;D2;D3;C$;S3$(KK):RETURN
6100 ONTYGOTO6110,6120,6130,6140,6150,6160
6110 LPRINTUSINGF$;ABS(A1$(KK));C$;A3$(KK);A4$(KK);P;R#:RETURN
6120 LPRINTUSINGF$;C1$(KK);D2;D3;C$;C4$(KK);ABS(A1$(C5$(KK))):RETURN

```

```

6130 LPRINTUSINGF$; C1%(KK); DY; YN; C$; C4(KK); ABS(A1%(C5%(KK))); D2; D3: RETURN
6140 LPRINTUSINGF$; D2; D3; C$; E3(KK); ABS(A1%(E4%(KK))); RETURN
6150 LPRINTUSINGF$; D2; D3; C$; D3(KK); RETURN
6160 LPRINTUSINGF$; D2; D3; C$; S3(KK); RETURN
9400 POKEI1, I2: RETURN
9500 JJ=J: J=1: GOSUB400: J=JJ: PRINT: I1=15423: I2=32
9505 IO=0: IFDK=1THENDV=1: ONERRORGOTO9900
9510 IFDK=1THENLINEINPUT"FILESPEC OF INPUT FILE: "; B$: OPEN"I", DV, B$: ONERRORGOTO0
9520 IFDK=0THENDV=-1: INPUT"LOAD DATA TAPE(PLAY), THEN PRESS ENTER"; A$
9530 INPUT#DV, D: PRINT"DATE OF DATA: "; : GOSUB950: PRINT: PRINT"READING ACCOUNTS
9535 IFF=1THEN9660
9540 INPUT#DV, K1: IFK1+1>M1THEN9700
9550 IFK1<>0FORI=1TOK1+1: INPUT#DV, A1%(I), A2$(I), A3$(I), A4$(I), A6%(I): GOSUB9400: N
EXT
9560 PRINT"READING CHECKS": INPUT#DV, K2: IFK2>M2THEN9700
9570 IFK2<>0FORI=1TOK2: INPUT#DV, C1%(I), C2$(I), C3$(I), C4(I), C5%(I), C6%(I): GOSUB94
00: NEXT
9580 PRINT"READING CASH EXPENSES": INPUT#DV, K3: IFK3>M3THEN9700
9590 IFK3<>0FORI=1TOK3: INPUT#DV, E1%(I), E2$(I), E3(I), E4%(I): GOSUB9400: NEXT
9600 PRINT"READING DEPOSITS": INPUT#DV, K4: IFK4>M4THEN9700
9610 IFK4<>0FORI=1TOK4: INPUT#DV, D1%(I), D2$(I), D3(I): GOSUB9400: NEXT
9620 PRINT"READING SAVINGS DATA": INPUT#DV, K5: IFK5>M5THEN9700
9630 IFK5<>0THENFORI=1TOK5: INPUT#DV, S1%(I), S2$(I), S3(I): GOSUB9400: NEXT
9640 F=1: IFDK=1THENCLOSE: RETURNELSERETURN
9660 INPUT#DV, K: IFK=0THEN9680
9665 FORI=1TOK: INPUT#DV, NN, A2$, A3$, A4$, A6%: GOSUB9400: IFI<4NEXT
9670 NN=ABS(NN): GOSUB450: IFII=0NEXT: GOT09680
9675 A3$(II)=A3$(II)+A3$: A4$(II)=A4$(II)+A4$: NEXT
9680 GOSUB320: F=1: IFDK=1THENCLOSE: RETURNELSERETURN
9700 E=2: GOSUB5000: GOSUB320: RETURN
9900 I=ERR/2+1: IFI=54THENE=1: GOSUB5000: RESUME9510
9910 PRINT: PRINT"UNEXPECTED ERROR #"I"IN LINE"ERL"-- RUN ABORTED": END

```



### IMPORTANT NOTICE

ALL RADIO SHACK COMPUTER PROGRAMS ARE DISTRIBUTED ON AN "AS IS" BASIS WITHOUT WARRANTY

Radio Shack shall have no liability or responsibility to customer or any other person or entity with respect to any liability, loss or damage caused or alleged to be caused directly or indirectly by computer equipment or programs sold by Radio Shack, including but not limited to any interruption of service, loss of business or anticipatory profits or consequential damages resulting from the use or operation of such computer or computer programs.

NOTE: Good data processing procedure dictates that the user test the program, run and test sample sets of data, and run the system in parallel with the system previously in use for a period of time adequate to insure that results of operation of the computer or program are satisfactory.

RADIO SHACK  A DIVISION OF TANDY CORPORATION

U.S.A.: FORT WORTH, TEXAS 76102

CANADA: BARRIE, ONTARIO L4M 4W5

---

#### TANDY CORPORATION

##### AUSTRALIA

280-316 VICTORIA ROAD  
RYDALMERE, N.S.W. 2116

##### BELGIUM

PARC INDUSTRIEL DE NANINNE  
5140 NANINNE

##### U. K.

BILSTON ROAD WEDNESBURY  
WEST MIDLANDS WS10 7JN

PRINTED IN U.S.A.

5/79